Google Web Images Groups News Froogle Local[New!] more »
xml configuration management Search Advanced Search Preferences

Web Results **1** - **10** of about **2,420,000** for **xml configuration management**. (**0.22** seconds)

The Code Project - **Configuration Management** for Smart Device ...
... **configuration** settings of a particular **configuration** section, the ... which returns true if the config section handlers ... which reads the SmartConfig.**xml** and stores ...
www.codeproject.com/netcf/SmartConfig.asp - 40k - Feb 21, 2005 - Cached - Similar pages

The Code Project - ASP.NET **configuration management**: an ...
... **configuration** settings **management** technique used nowadays store **configuration** values in the web.config file. This is an **XML configuration** file, containing ...
www.codeproject.com/useritems/WebSettingsAlt.asp - 38k - Cached - Similar pages
[More results from www.codeproject.com]

patterns & practices: **Configuration Management** Application Block ...
... Using 24% of available space. **XML** Releases. **Configuration** Manager (07/02/2003 • 992k • 2077 downloads) store config info where you want, encrypted or not ...
www.gotdotnet.com/Community/Workspaces/ Workspace.aspx?id=01875f69-9358-437b-a8ae-fa4bf2e3080f - 41k - Cached - Similar pages

EKJ Online - Ant: **XML**-based **configuration management** and scripting ...
... September 2002. Journal cover image Ant: **XML**-based **configuration management** and scripting. by Sean Kelly. Operating System(s): <none ...
www.elementkjournals.com/ premier/showArticle.asp?aid=9526 - 35k - Cached - Similar pages

NETWORKING JOB: Software **Configuration Management** Engineer ...
... automated build and deployment scripts with **XML configuration** based tools ... and ensure that all artifacts are properly placed under **configuration management**. ...
www.network-careers.com/showjob.cfm/200450 - 22k - Cached - Similar pages

XML Configuration(xmlconf) Bof
... for **XML** Network **Configuration** http://www.ietf.org/internet-dafts/draft-wasserman-xmlcomf-req-00.txt Towards **XML** Based **Management** and **Configuration** http://www ...
www.ietf.org/proceedings/02jul/168.htm - 10k - Cached - Similar pages

AdventNet Solutions for EMS/ NMS Solutions
... With it's approach of using discrete paramaterized **XML** documents to drive **configuration** activity, Web NMS **configuration management** makes it easier and less ...
www.adventnet.com/solutions/nevs_config.html - 24k - Feb 20, 2005 - Cached - Similar pages

[PDF] Logger **Configuration Management** Software - Functional ...
File Format: PDF/Adobe Acrobat - View as HTML
... This record contains the basic **configuration**, plus some status information about the logger: ... **XML** Fields ¡ <LOGGER_UPLOAD><CONFIG><LOGGER><ID> (upload) ...
chuck-wright.com/wcsdocs/cfg_mgmt.pdf - Similar pages

Managing Configuration Changes
... WebLogic Server imposes a change **management** process that ... file system by a set of

XML configuration files, centralized in the config.**xml** file, and ...
e-docs.bea.com/wls/docs90/domain_config/changes.html - 23k - Cached - Similar pages

[PDF] Open Configurator (Unified **Configuration Management** on Unix based ...
File Format: PDF/Adobe Acrobat - View as HTML
... information for every registered service under the config- urator umbrella. ... software in order to represent all the **configuration** info in **XML** format must ...
www.inaccessnetworks.com/ian/projects/ openrouter/project/docs/configurator-v1.pdf - Similar pages

Goooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

xml configuration management Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

Understanding Domain Configuration

Note: Because this information documents a Beta release of a BEA product, it may be incomplete and subject to change.

Managing Configuration Changes

To provide a secure, predictable means for distributing configuration changes in a domain, WebLogic Server imposes a change management process that loosely resembles a database transaction. The configuration of a domain is represented on the file system by a set of XML configuration files, centralized in the `config.xml` file, and at runtime by a hierarchy of Configuration MBeans. When you edit the domain configuration, you edit a separate hierarchy of Configuration MBeans that resides on the Administration Server. To start the edit process, you obtain a lock on the edit hierarchy to prevent other people from making changes. When you finish making changes, you save the changes. The changes do not take effect, however, until you activate them, distributing them to all server instances in the domain. When you activate changes, each server determines whether it can accept the change. If all servers are able to accept the change, they update their working configuration hierarchy and the change is completed.

Note that WebLogic Server's change management process applies to changes in domain and server configuration data, not to security or application data.

For more detailed information about how configuration changes are carried out through JMX and Configuration MBeans, see Understanding WebLogic Server MBeans in *Developing Manageable Applications with JMX*.

As described in Using WebLogic Tools to Configure a Domain, you can use a variety of different WebLogic Server tools to make configuration changes:

- Administration Console
- WebLogic Scripting Tool
- JMX APIs

Whichever tool you use to make configuration changes, WebLogic Server handles the change process in basically the same way.

The following sections describe configuration change management:

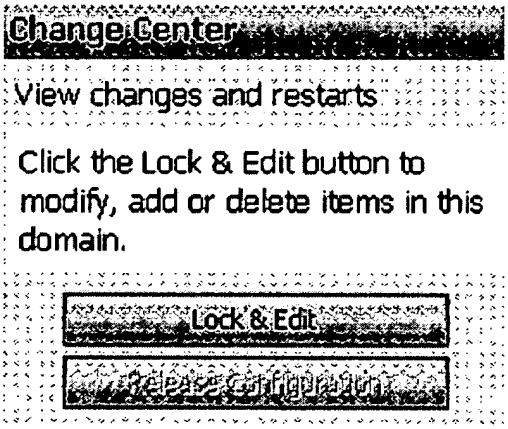
- Change Management in the Administration Console
- Configuration Change Management Process

- Configuration Management State Diagram

Change Management in the Administration Console

The WebLogic Administration Console centralizes the configuration change management process in the Change Center region:

Figure 5-1 Change Center



If you want to use the Administration Console to make configuration changes, you must first click the Lock & Edit button in the Change Center. When you click Lock & Edit, you obtain a lock on the editable hierarchy of Configuration MBeans for all servers in the domain (the edit tree).

As you make configuration changes using the Administration Console, you click Save (or in some cases Finish) on the appropriate pages. This does not cause the changes to take effect immediately; instead, when you click Save, you are saving the change to the edit tree and to the *domain-name*/pending/config.xml file and related configuration files. The changes take effect when you click Activate Changes in the Change Center. At that point, the configuration changes are distributed to each of the servers in the domain. If the changes are acceptable to each of the servers, then they take effect. If any server cannot accept a change, then all of the changes are rolled back from all of the servers in the domain. The changes are left in a pending state; you can then either edit the pending changes to resolve the problem or revert the pending changes.

Configuration Change Management Process

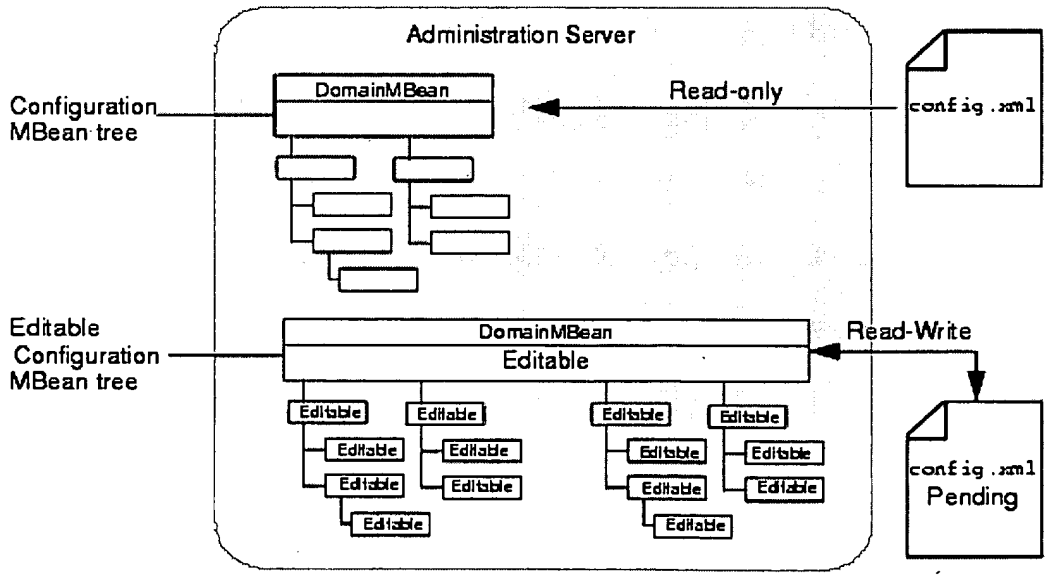
The following steps describe the process in detail, starting from when you first boot the domain's Administration Server:

1. When the Administration Server starts, it reads the domain's configuration files, including `config.xml` file and any subsidiary configuration files referred to by the `config.xml` file and uses the data to instantiate the following MBean trees:
   - A read-only tree of Configuration MBeans that contains the current configuration of resources that are on the Administration Server.
   - An editable tree of all Configuration MBeans for all servers in the domain.

     Note: The Administration Server also instantiates a Runtime MBean tree and a DomainRuntime MBean tree, but these are not used for configuration management.
2. To initiate a configuration change, you do the following:
   a. Obtain a lock on the current configuration.
   b. Make any changes you desire, using the tool of your choice (the Administration Console, WLST, the JMX APIs, etc.)
   c. Save your changes to a pending version of the `config.xml` file.
3. The Configuration Manager service saves all data from the edit MBean tree to a separate set of configuration files in a directory named `pending`. See Figure 5-2.

   The `pending` directory is immediately below the domain's root directory. For example, if your domain is named `mydomain`, then the default pathname of the pending `config.xml` file is `mydomain/pending/config.xml`.

Figure 5-2 The Administration Server's Pending config.xml File



4. Make additional changes or undo changes that you have already made.
5. When you are ready, activate your changes in the domain, using the Activate Changes button in the Administration Console's Change Center or using the `ConfigurationManagerMBean`.

   When you activate changes (see Figure 5-3):

   a. For each server instance in the domain, the Configuration Manager service copies the pending configuration files to a `pending` directory in the server's root directory.

      If a Managed Server shares its root directory with the Administration Server, `ConfigurationManagerMBean` does not copy the pending configuration files; the Managed Server uses the Administration Server's pending file.

   b. Each server instance compares its current configuration with the configuration in the pending file.

   c. Subsystems within each server vote on whether they can consume the new configuration.

      If any subsystem indicates that it cannot consume the changes, the entire activation process is rolled back and the `ConfigurationManagerMBean` emits an exception. You can modify your changes and retry the change activation, or you can abandon your lock, in which case the edit Configuration MBean tree and the pending configuration files are reverted to the configuration in the read-only Configuration MBean tree and configuration files.

   d. If all subsystems on all servers can consume the change, the Configuration Manager service replaces the read-only configuration files on each server instance in the domain with the pending configuration files.

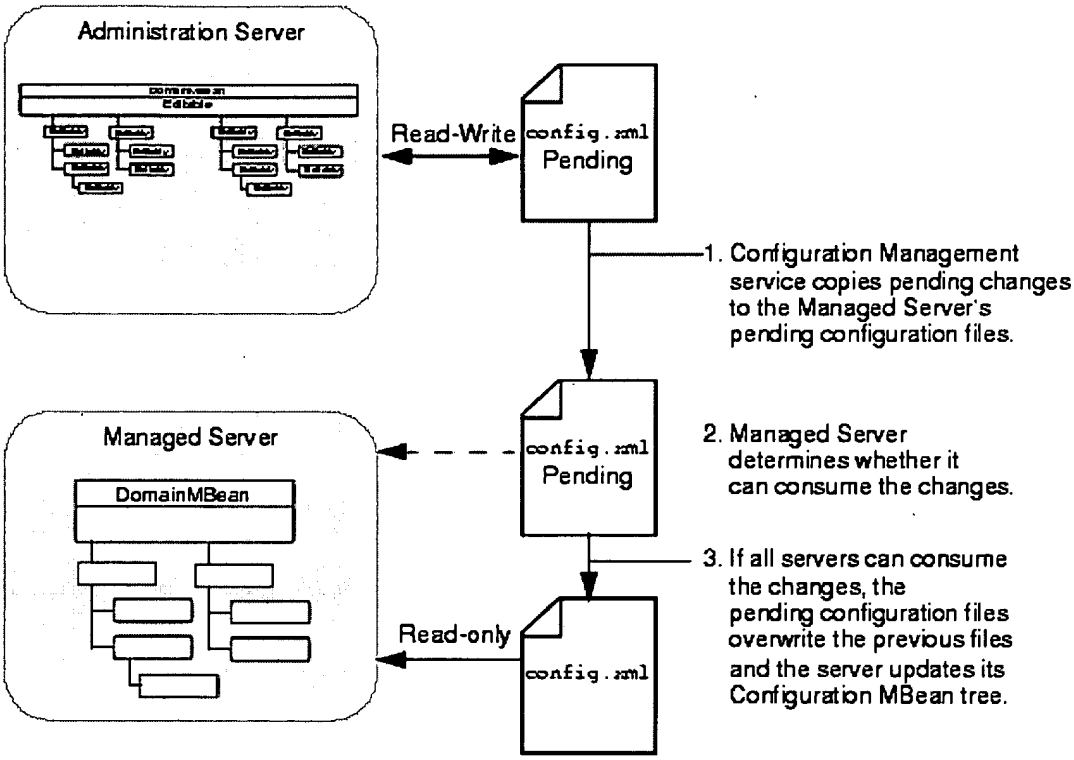
   e. Each server instance updates its beans and its read-only Configuration MBean tree according to the changes in the new configuration files.

   f. The pending configuration files are then deleted from the `pending` directory.

6. You can retain your lock to make additional changes or release it so that others can update the configuration. You can configure a timeout period that causes the Configuration Manager service to abandon a lock.

Note: The configuration change lock does not prevent you from making conflicting configuration edits using the same administrator user account. For example, if you obtain a configuration change lock using the Administration Console, and then use the WebLogic Scripting Tool with the same user account, you will access the same edit session that you opened in the Administration Console and you will not be locked out of making changes with the Scripting Tool. Since this can lead to confusion and conflicting configuration changes, this is not a recommended practice. You can reduce the risk that such a situation might occur by maintaining separate administrator user accounts for each person with an administrative role. Similar problems can still occur, however, if you have multiple instances of the same script using the same user account.

Figure 5-3 Activating Changes in Managed Servers



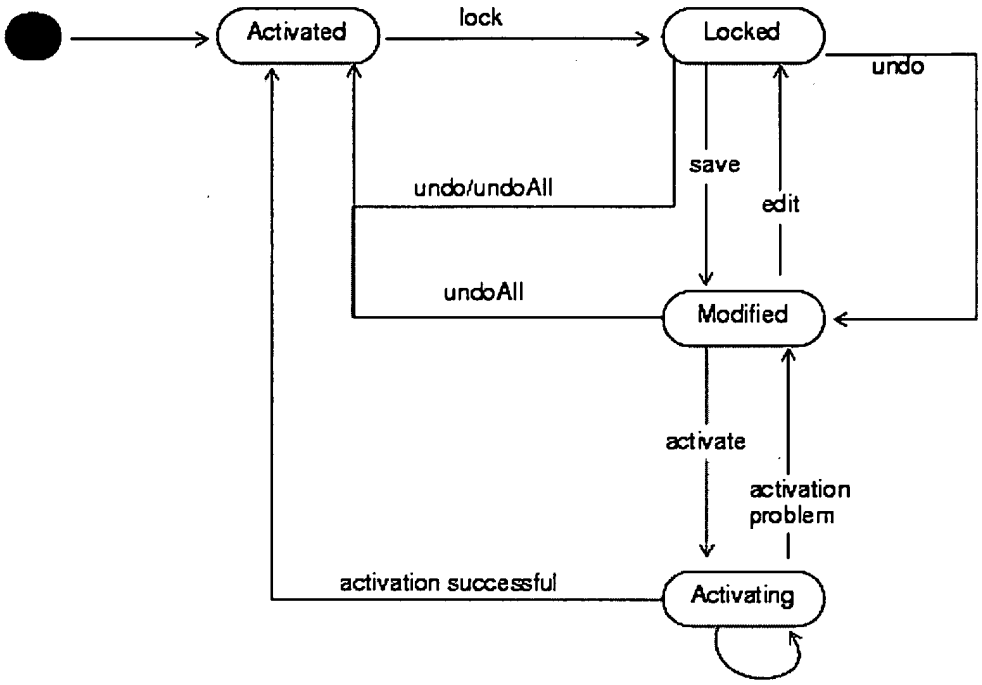
Resolving Change Conflicts

In the event that you have saved more than one change set without activating them and one change would invalidate a prior change, the Change Management service requires you to manually resolve the invalidation before it will save your changes.

Configuration Management State Diagram

The Configuration Management service follows a series of states, which are described in Figure 5-4.

Figure 5-4 Configuration Management State Diagram



back to top previous next

Terms used **xml configuration management** Found **19,676** of **150,885**

Sort results by: relevance
Display results: expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Versioning and fragmentation: Fine-grained, structured configuration management for web projects
Tien Nhut Nguyen, Ethan Vincent Munson, Cheng Thao
May 2004 **Proceedings of the 13th international conference on World Wide Web**
Full text available: pdf(698.15 KB) Additional Information: full citation, abstract, references, index terms

Researchers in Web engineering have regularly noted that existing Web application development environments provide little support for managing the evolution of Web applications. Key limitations of Web development environments include line-oriented change models that inadequately represent Web document semantics and in ability to model changes to link structure or the set of objects making up the Webapplication. Developers may find it difficult to grasp how theoverall structure of the Web applica ...

Keywords: software configuration management, version control, web engineering

2 An analysis of XML database solutions for the management of MPEG-7 media descriptions
Utz Westermann, Wolfgang Klas
December 2003 **ACM Computing Surveys (CSUR)**, Volume 35 Issue 4
Full text available: pdf(448.76 KB) Additional Information: full citation, abstract, references, index terms

MPEG-7 constitutes a promising standard for the description of multimedia content. It can be expected that a lot of applications based on MPEG-7 media descriptions will be set up in the near future. Therefore, means for the adequate management of large amounts of MPEG-7-compliant media descriptions are certainly desirable. Essentially, MPEG-7 media descriptions are XML documents following media description schemes defined with a variant of XML Schema. Thus, it is reasonable to investigate curren ...

Keywords: MPEG-7, XML database systems, multimedia databases

3 Posters: Multimedia streaming services: specification, implementation, and retrieval
Björn Althun, Martin Zimmermann
November 2003 **Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval**
Full text available: pdf(263.35 KB) Additional Information: full citation, abstract, references, index terms

The central purpose of this paper is to present a novel framework supporting the specification, the implementation and retrieval of media streaming services. It provides an integrated service development environment comprising of a streaming service model, a service specification language and several implementation and retrieval tools. Our approach is based on a clear separation of a streaming service specification, and its implementation by a distributed application and can be used for differen ...

Keywords: XML, multimedia retrieval, streaming service

4 SCM-10: tenth international workshop on software configuration management new practices, new challenges, and new boundaries

André van der Hoek

July 2001 **Proceedings of the 23rd International Conference on Software Engineering**

Full text available: pdf(46.83 KB) Additional Information: full citation, abstract, index terms
Publisher Site

With its tenth anniversary, the SCM workshop series is "leaving behind" a long and successful past: although the tools, techniques, and processes developed over the last twenty years have proven to be extremely valuable, they are inadequate to address the rapidly changing practices that define the world of tomorrow. Exemplified by component-based software development, open source, virtual enterprises, a ubiquitous use of XML, and the increasing dynamic nature of software, these ...

5 Hypertext versioning: The molhado hypertext versioning system

Tien N. Nguyen, Ethan V. Munson, John T. Boyland

August 2004 **Proceedings of the fifteenth ACM conference on Hypertext & hypermedia**

Full text available: pdf(943.36 KB) Additional Information: full citation, abstract, references, index terms

This paper describes *Molhado*, a hypertext versioning and software configuration management system that is distinguished from previous systems by its flexible product versioning and structural configuration management model. The model enables a unified versioning framework for atomic and composite software artifacts, and hypermedia structures among them in a fine-grained manner at the logical level. Hypermedia structures are managed separately from documents' contents. Molhado explicitly r ...

Keywords: hypertext versioning, software configuration management, software engineering, version control

6 Mae---a system model and environment for managing architectural evolution

Roshanak Roshandel, André Van Der Hoek, Marija Mikic-Rakic, Nenad Medvidovic

April 2004 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 13 Issue 2

Full text available: pdf(2.39 MB) Additional Information: full citation, abstract, references, index terms

As with any other artifact produced as part of the software life cycle, software architectures evolve and this evolution must be managed. One approach to doing so would be to apply any of a host of existing configuration management systems, which have long been used successfully at the level of source code. Unfortunately, such an approach leads to many problems that prevent effective management of architectural evolution. To overcome these problems, we have developed an alternative approach cent ...

Keywords: Design environment, Mae, evolution, system model

7 Traffic generation and analysis: Self-configuring network traffic generation
Joel Sommers, Paul Barford
October 2004 **Proceedings of the 4th ACM SIGCOMM conference on Internet measurement**
Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, index terms

The ability to generate repeatable, realistic network traffic is critical in both simulation and testbed environments. Traffic generation capabilities to date have been limited to either simple sequenced packet streams typically aimed at throughput testing, or to application-specific tools focused on, for example, recreating representative HTTP requests. In this paper we describe Harpoon, a new application-independent tool for generating representative packet traffic at the <i>IP flow lev ...

Keywords: network flows, traffic generation

8 Special section on advanced XML data processing: XML document versioning
Shu Yao Chien, Vassilis J. Tsotras, Carlo Zaniolo
September 2001 **ACM SIGMOD Record**, Volume 30 Issue 3
Full text available: pdf(716.34 KB) Additional Information: full citation, abstract, references, index terms

Managing multiple versions of XML documents represents an important problem, because of many applications ranging from traditional ones, such as software configuration control, to new ones, such as link permanence of web documents. Research on managing multiversion XML documents seeks to provide efficient and robust techniques for (i) storing and retrieving, (ii) exchanging, and (iii) querying such documents. In this paper, we first show that traditional version control methods, such as RCS, and ...

9 Industry session 3: database performance and interface: Using specification-driven concepts for distributed data management and dissemination
M. Brian Blake
November 2002 **Proceedings of the eleventh international conference on Information and knowledge management**
Full text available: pdf(206.57 KB) Additional Information: full citation, abstract, references, index terms

At the MITRE Corporation-Center for Advanced Aviation System Development (CAASD), software engineers work closely with both analyst and domain experts to develop software simulations in the air traffic management domain. In this environment, software simulations are applications that take large amounts of real-world operational information, and through calculations, derivations, and display extends the original information to produce some new insight into the domain. This new insight or knowledg ...

Keywords: Java servlets, XML, XSL, data management

10 XML processing: Ctree: a compact tree for indexing XML data
Qinghua Zou, Shaorong Liu, Wesley W. Chu
November 2004 **Proceedings of the 6th annual ACM international workshop on Web information and data management**
Full text available: pdf(272.47 KB) Additional Information: full citation, abstract, references, index terms

In this paper, we propose a novel compact tree (Ctree) for XML indexing, which provides not only concise path summaries at the group level but also detailed child-parent links at the element level. Group level mapping allows efficient pruning of a large search space while element level mapping provides fast access to the parent of an element. Due to the tree nature of XML data and queries, such fast child-to-parent access is essential for efficient XML query processing. Using group-based elem ...

Keywords: Ctree, XML index, XQuery evaluation, path summary, value index

11 Key management and key exchange: A temporal key management scheme for secure broadcasting of XML documents

Elisa Bertino, Barbara Carminati, Elena Ferrari

November 2002 **Proceedings of the 9th ACM conference on Computer and communications security**

Full text available: pdf(242.89 KB) Additional Information: full citation, abstract, references, index terms

Secure broadcasting of web documents is becoming a crucial need for many web-based applications. Under the broadcast document dissemination strategy a web document source periodically broad-casts (portions of) its documents to a possibly large community of subjects, without the need of explicit subject requests. By secure broadcasting we mean that the delivery of information to sub-jects must obey the access control policies of the document source. Since different subjects may have the right to ...

Keywords: XML, secure broadcasting, temporal key management

12 Research sessions: potpourri: Workflow management with service quality guarantees

Michael Gillmann, Gerhard Weikum, Wolfgang Wonner

June 2002 **Proceedings of the 2002 ACM SIGMOD international conference on Management of data**

Full text available: pdf(1.29 MB) Additional Information: full citation, abstract, references, index terms

Workflow management systems (WFMS) that are geared for the orchestration of business processes across multiple organizations are complex distributed systems: they consist of multiple workflow engines, application servers, and communication middleware servers such as ORBs, where each of these server types can be replicated on multiple computers for scalability and availability.Finding an appropriate system configuration with guaranteed application-specific quality of service in terms of throughpu ...

13 Document querying and transformation: A three-way merge for XML documents

Tancred Lindholm

October 2004 **Proceedings of the 2004 ACM symposium on Document engineering**

Full text available: pdf(500.99 KB) Additional Information: full citation, abstract, references, index terms

Three-way merging is a technique that may be employed for reintegrating changes to a document in cases where multiple independently modified copies have been made. While tools for three-way merge of ASCII text files exist in the form of the ubiquitous diff and patch tools these are of limited applicability to XML documents.

We present a method for three-way merging of XML which is targeted at merging XML formats that model human-authored documents as ordered trees (e.g. rich text forma ...

Keywords: XML, collaborative editing, conflict, structured text, three-way merge

14 Web mining and clustering: A version model for supporting adaptation of web pages

Rodrigo Giacomini Moro, Renata de Matos Galante, Carlos Alberto Heuser

November 2004 **Proceedings of the 6th annual ACM international workshop on Web information and data management**

Full text available: pdf(627.34 KB) Additional Information: full citation, abstract, references, index terms

Maintenance of large Web sites is a complex task, similar in some sense to software maintenance. Content should be separated from the formatting rules, allowing independent development and maintenance of both parts. As in software maintenance, version management is important in order to separate stable versions from versions under development. Further a version model that allows alternative versions may be used to support adaptation and personalization of Web content and formatting. To fulfil ...

Keywords: configurations, hyperdocument model, versions, web adaptation

15 Archiving scientific data

Peter Buneman, Sanjeev Khanna, Keishi Tajima, Wang-Chiew Tan

March 2004 **ACM Transactions on Database Systems (TODS)**, Volume 29 Issue 1

Full text available: pdf(745.61 KB) Additional Information: full citation, abstract, references, index terms

Archiving is important for scientific data, where it is necessary to record all past versions of a database in order to verify findings based upon a specific version. Much scientific data is held in a hierachical format and has a key structure that provides a canonical identification for each element of the hierarchy. In this article, we exploit these properties to develop an archiving technique that is both efficient in its use of space and preserves the continuity of elements through versions ...

Keywords: Keys for XML

16 eXtreme deployment: distributing and configuring 450 student laptops in five hours

E. Axel Larsson, Russell Sprague

October 2004 **Proceedings of the 32nd annual ACM SIGUCCS conference on User services**

Full text available: pdf(471.82 KB) Additional Information: full citation, abstract, references, index terms

Drew University has been recognized as a leader in Ubiquitous Computing since 1984, with the introduction of our Computer Initiative program, distributing more than 450 notebook computers to incoming students.

Drew's first Windows XP implementation did not register student computers into an Active Directory domain, instead opting for an unmanaged local account, which caused many support issues. By contract, having the computers in the domain would mean that users would not need a sepa ...

Keywords: Microsoft, PHP, XML-RPC, active directory, apache, deployment, imaging, ubiquitous computing, windows

17 Single sourcing: Beyond theory: making single-sourcing actually work

Liz Fraley

October 2003 **Proceedings of the 21st annual international conference on Documentation**

Full text available: pdf(230.43 KB) Additional Information: full citation, abstract, references, index terms

In this paper, I discuss how we made single-sourcing work at Juniper Networks. This is a practical discussion of issues, problems, and successes.

Keywords: Arbortext, E3, Epic, FrameMaker, Interwoven, Java, Juniper Networks, TeamSite, WebWorks, XML, XSLT, branching, case study, chunking, commit, documentation, modular writing, publishing, single source, single-sourcing

18 Managing the process: The software concordance: a new software document management environment

Tien N. Nguyen, Ethan V. Munson

October 2003 **Proceedings of the 21st annual international conference on Documentation**

Full text available: pdf(177.42 KB) Additional Information: full citation, abstract, references, index terms

In this paper, we describe the efforts of Juniper Networks to implement a Feature Guide documentation manual and discuss the usability merits of this documentation method.

Keywords: documentation, hypermedia, software engineering

19 A vision for management of complex models

Phillip A. Bernstein, Alon Y. Halevy, Rachel A. Pottinger

December 2000 **ACM SIGMOD Record**, Volume 29 Issue 4

Full text available: pdf(907.42 KB) Additional Information: full citation, abstract, citings, index terms

Many problems encountered when building applications of database systems involve the manipulation of models. By "model," we mean a complex structure that represents a design artifact, such as a relational schema, object-oriented interface, UML model, XML DTD, web-site schema, semantic network, complex document, or software configuration. Many uses of models involve managing changes in models and transformations of data from one model into another. These uses require an explicit representation of ...

20 Software engineering and middleware: Integrating publisher/subscriber services in component middleware for distributed real-time and embedded systems

George T. Edwards, Douglas C. Schmidt, Aniruddha Gokhale

April 2004 **Proceedings of the 42nd annual Southeast regional conference**

Full text available: pdf(444.01 KB) Additional Information: full citation, abstract, references, index terms

Although component-based software development has widespread acceptance in the enterprise business and desktop application domains, developers of distributed real-time and embedded (DRE) systems have encountered limitations with the available component middleware platforms, such as the CORBA Component Model (CCM) and the Java 2 Enterprise Edition (J2EE). These limitations often preclude developers of DRE systems from fully exploiting the benefits of component software. In particular, component m ...

Keywords: CORBA Component Model, component middleware, model-based systems, real-time event service

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Terms used **xml configuration management** Found **19,676** of **150,885**

Sort results by relevance Save results to a Binder

Display results expanded form Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 21 - 40 of 200 Result page: previous 1 **2** 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

21 Invited workshop on middleware interoperability of enterprise applications: An open system architecture for operation support system at telecommunications service providers

Cledson Akio Sakurai, Moacyr Martucci Junior

September 2003 **Proceedings of the 1st international symposium on Information and communication technologies**

Full text available: pdf(142.00 KB) Additional Information: full citation, abstract, references

To adapt at the new market, telecommunications service providers (carriers) need to supply a larger variety of low cost and high availability services, as well as speed in the development and delivery new value added services and control. Therefore, carriers need to implement Operation Support Systems (OSS) able to attend the customer's new needs and market opportunities quickly, but preserving the existing infrastructure of network elements and management systems. Thus, the market requires a sy ...

Keywords: OSS, distributed Architecture, middleware, plug and play, telecommunication

22 B2B contract implementation using windows DNS

Ning He, Zoran Milosevic

January 2001 **Australian Computer Science Communications**, Volume 23 Issue 6

Full text available: pdf(863.94 KB) Publisher Site Additional Information: full citation, abstract, references

This paper describes our implementation of a support infrastructure for electronic contracting --- an important ingredient of Business-to-Business (B2B) e-commerce. The paper first explains the main benefits of the new generation of Microsoft technologies - Windows Distributed interNet Applications Architecture (DNA) and BizTalk. This is followed by a detailed description of how we take advantage of the XML tools provided by these technologies - to implement our enterprise model of contracts. We ...

Keywords: BizTalk, XML-message, business contract

23 Differences between versions of UML diagrams

Dirk Ohst, Michael Welle, Udo Kelter

September 2003 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 9th**

European software engineering conference held jointly with 11th ACM SIGSOFT international symposium on Foundations of software engineering, Volume 28 Issue 5

Full text available: pdf(202.32 KB) Additional Information: full citation, abstract, references, index terms

This paper addresses the problem of how to detect and visualise differences between versions of UML documents such as class or object diagrams. Our basic approach for showing the differences between two documents is to use a unified document which contains the common and specific parts of both base documents; the specific parts are highlighted. The main problems are (a) how to abstract from modifications done to the layout and other (document type-specific) details which are considered irrelevan ...

Keywords: UML diagrams, configuration, design transaction, differences, fine-grained data model, software engineering environments, versions

24 A secure execution framework for Java

Manfred Hauswirth, Clemens Kerer, Roman Kurmanowytsch

November 2000 **Proceedings of the 7th ACM conference on Computer and communications security**

Full text available: pdf(430.90 KB) Additional Information: full citation, references, citings, index terms

Keywords: Java security management, XML-based security configuration, management GUIs

25 CHIME: a metadata-based distributed software development environment

Stephen E. Dossick, Gail E. Kaiser

October 1999 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 7th European software engineering conference held jointly with the 7th ACM SIGSOFT international symposium on Foundations of software engineering**, Volume 24 Issue 6

Full text available: pdf(940.46 KB) Additional Information: full citation, abstract, references, citings, index terms

We introduce CHIME, the Columbia Hypermedia IMmersion Environment, a metadata-based information environment, and describe its potential applications for internet and intranet-based distributed software development. CHIME derives many of its concepts from Multi-User Domains (MUDs), placing users in a semi-automatically generated 3D virtual world representing the software system. Users interact with project artifacts by "walking around" the virtual world, where they potentially en ...

26 Efficient schemes for managing multiversionXML documents

S.-Y. Chien, V. J. Tsotras, C. Zaniolo

December 2002 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 11 Issue 4

Full text available: pdf(926.90 KB) Additional Information: full citation, abstract, index terms

Multiversion support for XML documents is needed in many critical applications, such as software configuration control, cooperative authoring, web information warehouses, and "e-permanence" of web documents. In this paper, we introduce efficient and robust techniques for: (i) storing and retrieving; (ii) viewing and exchanging; and (iii) querying multiversion XML documents. We first discuss the limitations of traditional version control methods, such as RCS and SCCS, and then propose ...

Keywords: Historical queries, Temporal clustering, Temporal indexing, Version

management, XML database

27 Access control: First experiences using XACML for access control in distributed systems

Markus Lorch, Seth Proctor, Rebekah Lepro, Dennis Kafura, Sumit Shah
October 2003 **Proceedings of the 2003 ACM workshop on XML security**

Full text available: pdf(459.30 KB) Additional Information: full citation, abstract, references, citings, index terms

Authorization systems today are increasingly complex. They span domains of administration, rely on many different authentication sources, and manage permissions that can be as complex as the system itself. Worse still, while there are many standards that define authentication mechanisms, the standards that address authorization are less well defined and tend to work only within homogeneous systems. This paper presents XACML, a standard access control language, as one component of a distributed a ...

Keywords: access control decision, access control enforcement, authorization, distributed system security, policy language, policy management

28 E-Design Based on the Reuse Paradigm

L. Ghanmi, A. Ghrab, M. Hamdoun, B. Missaoui, K. Skiba, G. Saucier
March 2002 **Proceedings of the conference on Design, automation and test in Europe**

Full text available: pdf(244.78 KB) Publisher Site Additional Information: full citation, abstract

This paper gives an overview on a Virtualelectronic component or IP (Intellectual Property) exchange infrastructure whose main components area XML "well structured IP e-catalog Builder ¿"and a" XML IP profiler¿ While the first module is ae_publishing and an exchange management modulethe second has as role to extract from the designdirectories the IP files and to trigger their transferto the user site possibly via an IP distribution serverunder the catalog control. Direct Design fileextraction fro ...

29 Broadcast and on-line cultural heritage: Copyright protection and management and a web based library for digital images of the Hellenic cultural heritage

Dimitris K. Tsolis, George K. Tsolis, Emmanouil G. Karatzas, Theodore S. Papatheodorou
November 2001 **Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage**

Full text available: pdf(358.69 KB) Additional Information: full citation, abstract, references, index terms

The main issue addressed in this paper is the design and implementation of an Advanced Digital Image Repository, which offers specialized services and a Dedicated User Interface for the protection and management of the Intellectual Property Rights of digitized material. In addition, another main research area of this contribution is the implementation of a Web Based Library, supported by advanced technologies, for the proper presentation of the digital cultural content. The work described in thi ...

Keywords: copyright protection, databases, digital web archives, information systems, java applets, watermarking

30 A lightweight, message-oriented application server for the WWW

Ralf-Dieter Schimkat, Stefan Müller, Wolfgang Küchlin
March 2000 **Proceedings of the 2000 ACM symposium on Applied computing**

Full text available: pdf(1.37 MB) Additional Information: full citation, references, index terms

Keywords: application integration, application server, message bus, uniform Web interface

31 Semantic management of middleware

Daniel Oberle

October 2004 **Proceedings of the 1st international doctoral symposium on Middleware**

Full text available: pdf(112.87 KB) Additional Information: full citation, abstract, references, index terms

The Ph.D. proposal addresses the complexity of building distributed applications and systems with Application Servers and Web Services middleware, respectively. Despite their flexible XML-based configuration, taming the ever growing complexity remains all but an easy task. To remedy such problems, the thesis proposes an ontology-based approach to support the management (i.e. development and administration) of Application Server and Web Services based applications. The ontology captures proper ...

Keywords: application server, middleware, ontology, semantic technology, service oriented architecture, web service

32 Web-based tools, systems and environments: Software configuration, distribution, and deployment of web-services

Rainer Anzböck, Schahram Dustdar, Harald Gall

July 2002 **Proceedings of the 14th international conference on Software engineering and knowledge engineering**

Full text available: pdf(519.92 KB) Additional Information: full citation, abstract, references, index terms

Web-Services can be seen as a newly emerging distributed computing model for the Web. They cater for the need to establish business-to-business (B2B) interactions on the Web. Web-Services consider a loosely coupled component model encapsulating business logic and interact with other components using XML protocols. Based on one case study, this paper discusses architectural issues and requirements for software configuration, distribution, and deployment of web-services.

Keywords: software architecture, software distribution environments, web-services

33 Research sessions: new styles of XML: Data stream management for historical XML data

Sujoe Bose, Leonidas Fegaras

June 2004 **Proceedings of the 2004 ACM SIGMOD international conference on Management of data**

Full text available: pdf(207.19 KB) Additional Information: full citation, abstract, references

We are presenting a framework for continuous querying of time-varying streamed XML data. A continuous stream in our framework consists of a finite XML document followed by a continuous stream of updates. The unit of update is an XML fragment, which can relate to other fragments through system-generated unique IDs. The reconstruction of temporal data from continuous updates at a current time is never materialized and historical queries operate directly on the fragmented streams. We are incorporat ...

34 Session 4: Web service applications: Dynamically authorized role-based access control for secure distributed computation

C. Joncheng Kuo, Polar Humenn
November 2002 **Proceedings of the 2002 ACM workshop on XML security**

Full text available: pdf(171.18 KB) Additional Information: full citation, abstract, references, index terms

This paper presents a mechanism for using the Object Management Group's Common Secure Interoperability Version 2 (CSIv2), Authorization Token Layer Acquisition Service (ATLAS), and XML security standards such as Security Assertion Markup Language (SAML) to develop role-based access control (RBAC) in a secure distributed computation system.The need for RBAC became evident in this kind of system because the components of the system are configured dynamically in specific neighbor relationships to e ...

Keywords: CORBA, Role-based access control, XML-based security assertions, attribute certificates, authorization domain

35 Web and e-business application: A Java based XML browser for consumer devices
Petri Vuorimaa, Teemu Ropponen, Niklas von Knorring, Mikko Honkala
March 2002 **Proceedings of the 2002 ACM symposium on Applied computing**

Full text available: pdf(918.25 KB) Additional Information: full citation, abstract, references, citings, index terms

Next generation consumer devices will all have an Internet connection. Thus, one vision is that the future multimedia services will be browser based. Extensible Markup Language (XML) is the most likely markup language. In this paper, we introduce a Java based XML browser called X-Smiles. It is intended for consumer devices and supports multimedia services. The main advantage of the X-Smiles browser is that it supports most of the XML related specifications. Different XML based languages can be m ...

Keywords: SMIL, SVG, XML, XSL FO, multimedia

36 Supporting software engineering with open hypermedia
Kenneth M. Anderson
December 1999 **ACM Computing Surveys (CSUR)**

Full text available: pdf(25.38 KB) Additional Information: full citation, references, citings, index terms

Keywords: case study, open hypermedia, software engineering

37 Drudgery and deep thought
Gregory Crane, Robert F. Chavez, Anne Mahoney, Thomas L. Milbank, Jeffrey A. Rydberg-Cox, David A. Smith, Clifford E. Wulfman
May 2001 **Communications of the ACM**, Volume 44 Issue 5

Full text available: pdf(290.21 KB) html(33.91 KB) Additional Information: full citation, references, citings, index terms

38 Electronic document technology: Developing an XML framework for metadata system
Ruey-Shun Chen, Shien-Chiang Yu
September 2003 **Proceedings of the 1st international symposium on Information and communication technologies**

Full text available: pdf(383.59 KB) Additional Information: full citation, abstract, references

Metadata is the media to describe the management and storing of kind of knowledge and well-preserved objects. Thus, with the compatibility of all various metadata, we can integrate related knowledge and with unification and management of information system. This paper introduces a system using the XML framework to be compatible with various metadata schemas, using the DTDs of XML to define the system schema structure, allowing more than one DTDs to exist. Therefore, it meets the demand of proces ...

Keywords: DTD, XML, metadata, schema, system design

39 Web engineering: Configuration management in a hypermedia-based software development environment

Tien N. Nguyen, Ethan V. Munson, John T. Boyland

August 2003 **Proceedings of the fourteenth ACM conference on Hypertext and hypermedia**

Full text available: pdf(42.82 KB) Additional Information: full citation, abstract, references, index terms

Several researchers have explored the use of hypermedia technology in software development environments (SDEs). However, existing hypermedia-based SDEs have only limited support for the evolutionary aspects of software projects. On the other hand, commercial software configuration management systems (SCMs) have had noticeable success in helping developers manage system evolution. While researchers in the hypermedia community acknowledged the need for strong version control support in their syste ...

Keywords: configuration management, hypermedia, software engineering, version control

40 Effective Web data extraction with standard XML technologies

Jussi Myllymaki

April 2001 **Proceedings of the tenth international conference on World Wide Web**

Full text available: pdf(198.81 KB) Additional Information: full citation, references, citings, index terms

Keywords: crawling, data extraction, deep Web, semistructured data, wrappers

Software

March/April 1992 (Vol. 9, No. 2)

pp. 62-68 • A Hypertext Based Software-Engineering Environment

Jacob L. Cybulski, Karl Reed

HyperCASE, an architectural framework for integrating CASE tools under an extended hypertext system, is described. HyperCASE's objective is to provide a powerful, user-friendly, integrated development platform that can significantly raise productivity. Its specific goal is to support software developers in project management, system analysis, design, and coding. HyperCASE integrates tools by combining a hypertext-based user interface with a common knowledge-based document repository. It includes extensive natural-language capabilities tailored to the CASE domain. These are used in the interface to the software repository, providing an alternative to hypertext information management and interdocument navigation. English input can be analyzed during informal system-requirements specification, allowing a significant degree of automation for design and concept reuse at the earliest development stages. HyperCASE's three subsystems, HyperEdit, the graphical user interface, HyperBase, the knowledge base, and HyperDict, the data dictionary, are discussed.

Index Terms- tool integration; hypertext; software-engineering environment; HyperCASE; CASE tools; extended hypertext system; user-friendly, integrated development platform; project management; system analysis; design; coding; hypertext-based user interface; knowledge-based document repository; natural-language; software repository; English input; informal system-requirements specification; concept reuse; HyperEdit; graphical user interface; HyperBase; HyperDict; data dictionary; graphical user interfaces; hypermedia; knowledge based systems; natural languages; programming environments; project support environments; software tools

References

Reference Information

[1] R. Rock-Evans and B. Engelien, *Analysis Techniques for CASE: A Detailed Evaluation*, Ovum Ltd., London, 1989.

[2] R. Rock-Evans, *CASE Analyst Workbenches: A Detailed Product Evaluation*, Ovum Ltd., London, 1989.

[3] J. Conklin, "Hypertext: An Introduction and Survey," *Computer*, Vol. 20, No. 9, Sept. 1987, pp. 17-41.

[4] J. Bigelow, "Hypertext And CASE," *IEEE Software*, vol. 5, no. 2, pp. 23-27, 1988.

[5] P. Garg, "Abstraction Mechanisms in Hypertext," *Communications of the ACM*, vol. 31, no. 1, pp. 862-870, 1988-01-01.

[6] W. Scacchi, "The USC System Factory Project," *Proc. Software Symp.*, Software Engineers Assoc., ACM Press, New York, Jan. 1989, pp. 9-41.

[7] P. Carando, "Shadow: Fusing Hypertext With AI," *IEEE Expert: Intelligent Systems and Their Applications*, vol. 4, no. 4, pp. 65-78, 1989.

[8] P. Garg and W. Scacchi, "ISHYS: Designing an Intelligent Software Hypertext System," *IEEE Expert: Intelligent Systems and Their Applications*, vol. 4, no. 3, pp. 52-63, 1989.

[9] P. Puncello, et al., "ASPIS: A Knowledge-Based CASE Environment," *IEEE Software*, vol. 5, no. 2, pp. 58-65, 1988.

A Hypertext Based Software-Engineering Environment

Full text Publisher Site

Source **IEEE Software** archive
Volume 9 , Issue 2 (March 1992) table of contents
Pages: 62 - 68
Year of Publication: 1992
ISSN:0740-7459

Authors Jacob L. Cybulski
Karl Reed

Publisher IEEE Computer Society Press Los Alamitos, CA, USA

DOI Bookmark: 10.1109/52.120603

↑ ABSTRACT

HyperCASE, an architectural framework for integrating CASE tools under an extended hypertext system, is described. HyperCASE's objective is to provide a powerful, user-friendly, integrated development platform that can significantly raise productivity. Its specific goal is to support software developers in project management, system analysis, design, and coding. HyperCASE integrates tools by combining a hypertext-based user interface with a common knowledge-based document repository. It includes extensive natural-language capabilities tailored to the CASE domain. These are used in the interface to the software repository, providing an alternative to hypertext information management and interdocument navigation. English input can be analyzed during informal system-requirements specification, allowing a significant degree of automation for design and concept reuse at the earliest development stages. HyperCASE's three subsystems, HyperEdit, the graphical user interface, HyperBase, the knowledge base, and HyperDict, the data dictionary, are discussed.

↑ REFERENCES

Note: OCR errors may be found in this Reference List extracted from the full text article. ACM has opted to expose the complete List rather than only correct and linked references.

1 Rosemary Rock-Evans , Brigitte Engelien, Analysis techniques for CASE: a detailed evaluation, Ovum, London, UK, 1990

2 Rosemary Rock-Evans, CASE analyst workbenches: a detailed product evaluation, Ovum, London, UK, 1990

3 Jeff Conklin, Hypertext: an introduction and survey, Computer, v.20 n.9, p.17-41, Sept. 1987

4 James Bigelow, Hypertext and CASE, IEEE Software, v.5 n.2, p.23-27, March 1988

5 Pankaj,K. Garg, Abstraction mechanisms in hypertext, Communications of the ACM, v.31 n.7, p.862-870, July 1988

6 6. W. Scacchi, "The USC System Factory Project," Proc. Software Symp., Software Engineers Assoc., ACM Press, New York, Jan. 1989, pp. 9-41.

7 Patricia Carando, Shadow: Fusing Hypertext with AI, IEEE Expert: Intelligent Systems and Their Applications, v.4 n.4, p.65-78, December 1989

8 Pankaj K. Garg , Walt Scacchi, ISHYS: Designing an Intelligent Software Hypertext System, IEEE Expert: Intelligent Systems and Their Applications, v.4 n.3, p.52-63, September 1989

9 P. Paolo Puncello , Piero Torrigiani , Francesco Pietri , Riccardo Burlon , Bruno Cardile , Mirella Conti, ASPIS: A Knowledge-Based CASE Environment, IEEE Software, v.5 n.2, p.58-65, March 1988

↑ CITINGS *10*

Allison L. Powell , James C. French , John C. Knight, A systematic approach to creating and maintaining software documentation, Proceedings of the 1996 ACM symposium on Applied Computing, p.201-208, February 17-19, 1996, Philadelphia, Pennsylvania, United States

Diane Crawford, Technical correspondence, Communications of the ACM, v.36 n.11, p.18, Nov. 1993

Tien N. Nguyen , Ethan V. Munson , John T. Boyland, Configuration management in a hypermedia-based software development environment, Proceedings of the fourteenth ACM conference on Hypertext and hypermedia, August 26-30, 2003, Nottingham, UK

James C. Ferrans , David W. Hurst , Michael A. Sennett , Burton M. Covnot , Wenguang Ji , Peter Kajka , Wei Ouyang, HyperWeb: a framework for hypermedia-based environments, ACM SIGSOFT Software Engineering Notes, v.17 n.5, p.1-10, Dec. 1992

Harri Oinas-Kukkonen, Flexible CASE and hypertext, ACM Computing Surveys (CSUR), v.31 n.4es, Dec. 1999

Tien N. Nguyen , Ethan V. Munson, The software concordance: a new software document management environment, Proceedings of the 21st annual international conference on Documentation, October 12-15, 2003, San Francisco, CA, USA

Tien N. Nguyen , Ethan V. Munson , John T. Boyland, The molhado hypertext versioning system, Proceedings of the fifteenth ACM conference on Hypertext & hypermedia, August 09-13, 2004, Santa Cruz, CA, USA

Jukka Paakki , Jussi Koskinen , Airi Salminen, From relational program dependencies to hypertextual access structures, Nordic Journal of Computing, v.4 n.1, p.3-36, Spring 1997

J. Koskinen, Experimental evaluation of hypertext access structures, Journal of Software Maintenance: Research and Practice, v.14 n.2, p.83-108, March 2002

↑ INDEX TERMS

Keywords:
CASE tools, English input, HyperBase, HyperCASE, HyperDict, HyperEdit, coding, concept reuse, data dictionary, design, extended hypertext system, graphical user interface, graphical user interfaces, hypermedia, hypertext, hypertext-based user interface, informal system-requirements specification, knowledge based systems, knowledge-based document repository, natural languages, natural-language, programming environments, project management, project support environments, software repository, software tools, software-engineering environment, system analysis, tool integration, user-friendly, integrated development platform

↑ **Collaborative Colleagues:**

Jacob L. Cybulski:	Jeffrey C. Allen
	Anthony Kram
	Tanya Linden
	Ricki Maurici
	Ralph D. "Butch" Neal
	Karl Reed
	Pradipta K. Sarkar
Karl Reed:	Jason Baragry
	Victor R. Basili
	Oddur Benediktsson
	Jacob L. Cybulski
	Darren Dalcher
	Anne Hannington
	Tafline Murnane
	Mark Woodman
	Liqun Wu
	Lin Zucconi

Day : Monday
Date: 2/21/2005

PALM INTRANET

Time: 20:44:10

Inventor Information for 09/936174

Inventor Name	City	State/Country
NACCACHE, DAVID	PARIS	FRANCE
ROUSSEAU, LUDOVIC	AUBAGNE	FRANCE
GIRARD, PIERRE	LA CIOTAT	FRANCE

Appln Info | Contents | Petition Info | Atty/Agent Info | Continuity Data | Foreign Data

Search Another: Application# [ ] Search **or Patent#** [ ] Search

PCT / [ ] / [ ] Search **or PG PUBS #** [ ] Search

Attorney Docket # [ ] Search

Bar Code # [ ] Search

To go back use Back button on your browser toolbar.

Back to PALM | ASSIGNMENT | OASIS | Home page

Day : Monday
Date: 2/21/2005

PALM INTRANET

Time: 20:44:13

Inventor Name Search Result

Your Search was:

Last Name = NACCACHE
First Name = DAVID

Application#	Patent#	Status	Date Filed	Title	Inventor Name
08082228	5414772	150	06/23/1993	SYSTEM FOR IMPROVING THE DIGITAL SIGNATURE ALGORITHM	NACCACHE, DAVID
08094058	5502764	150	01/24/1994	METHOD, IDENTIFICATION DEVICE AND VERIFICATION DEVICE FOR IDENTIFICATION AND/OR PERFORMING DIGITAL SIGNATURE	NACCACHE, DAVID
08094931	Not Issued	166	07/16/1993	METHOD AND APPARATUS FOR ACCESS CONTROL AND/OR IDENTIFICATION	NACCACHE, DAVID
08121599	5461675	150	09/14/1993	APPARATUS AND METHOD FOR ACCESS CONTROL	NACCACHE, DAVID
08122716	5347581	150	09/15/1993	VERIFICATION PROCESS FOR A COMMUNICATION SYSTEM	NACCACHE, DAVID
08135491	5654891	150	10/13/1993	METHOD AND APPARATUS FOR CONTROLLING AND/OR LIMITING SPEED EXCESS BY DRIVERS	NACCACHE, DAVID
08135492	Not Issued	163	10/13/1993	CARD, CARD READER AND METHOD FOR PROTOCOL SELECTION	NACCACHE, DAVID
08135493	5434917	150	10/13/1993	UNFORGEABLE IDENTIFICATION DEVICE, IDENTIFICATION DEVICE READER AND METHOD OF IDENTIFICATION	NACCACHE, DAVID
08199213	5479511	150	07/08/1994	METHOD, SENDER APPARATUS AND RECEIVER APPARATUS FOR MODULO OPERATION	NACCACHE, DAVID

08211280	Not Issued	167	11/16/1994	METHOD AND APPARATUS FOR SECURE TRANSMISSION OF VIDEO SIGNALS	NACCACHE, DAVID
08211281	Not Issued	161	11/07/1994	METHOD AND APPARATUS FOR SECURE TRANSMISSION OF VIDEO SIGNALS	NACCACHE, DAVID
08347570	5625695	150	11/30/1994	PROCESS FOR GENERATING DSA SIGNATURES WITH LOW-COST PORTABLE APPARATUSES	NACCACHE, DAVID
08400310	5452357	150	03/06/1995	METHOD AND APPARATUS FOR ACCESS CONTROL AND/OR IDENTIFICATION	NACCACHE, DAVID
08765483	5910989	150	03/06/1997	METHOD FOR THE GENERATION OF ELECTRONIC SIGNATURES, IN PARTICULAR FOR SMART CARDS	NACCACHE, DAVID
08860832	5946397	150	09/24/1997	METHOD OF CRYPTOGRAPHY WITH PUBLIC KEY BASED ON THE DISCRETE LOGARITHM	NACCACHE, DAVID
08875331	6226382	150	09/24/1997	METHOD FOR IMPLEMENTING A PRIVATE-KEY COMMUNICATION PROTOCOL BETWEEN TWO PROCESSING DEVICES	NACCACHE, DAVID
09194980	6459791	150	08/24/1999	PUBLIC KEY CRYPTOGRAPHY METHOD	NACCACHE, DAVID
09308369	Not Issued	161	07/13/2000	METHOD FOR SIGNING AND/OR AUTHENTICATION ELECTRONIC MESSAGES	NACCACHE, DAVID
09377666	Not Issued	161	08/19/1999	A CRYPTOGRAPHIC SYSTEM COMPRISING AN ENCRYPTION AND DECRYPTION SYSTEM AND A KEY ESCROW STSTEM AND THE ASSOCIATED EQUIPMENT AND DEVICES	NACCACHE, DAVID
09434102	Not Issued	161	11/05/1999	PSEUDO-RANDOM GENERATOR BASED ON A HASH CODING FUNCTION FOR CRYPTOGRAPHIC SYSTEMS REQUIRING RANDOM DRAWING	NACCACHE, DAVID
09646564	6698662	150	12/18/2000	DEVICES FOR HIDING	NACCACHE,

				OPERATIONS PERFORMED IN A MICROPROCESSER CARD	DAVID
09763158	Not Issued	041	05/07/2001	METHOD FOR TESTING A RANDOM NUMBER SOURCE AND ELECTRONIC DEVICES COMPRISING SAID METHOD	NACCACHE, DAVID
09802968	Not Issued	041	03/12/2001	PROBABILISTIC DIGITAL SIGNATURE METHOD	NACCACHE, DAVID
09807614	Not Issued	030	07/11/2001	ELECTRONIC COMPONENT FOR MASKING EXECUTION OF INSTRUCTIONS OR DATA MANIPULATION	NACCACHE, DAVID
09936174	Not Issued	030	12/06/2001	METHOD FOR MONITORING A PROGRAM FLOW	NACCACHE, DAVID
09959944	Not Issued	030	02/20/2002	COUNTERMEASURE METHOD IN AN ELECTRONIC COMPONENT USING A DYNAMIC SECRET KEY CRYPTOGRAPHIC ALGORITHM	NACCACHE, DAVID
10031065	Not Issued	030	04/02/2002	METHOD FOR IMPROVING A RANDOM NUMBER GENERATOR TO MAKE IT MORE RESISTANT AGAINST ATTACKS BY CURRENT MEASURING	NACCACHE, DAVID
10048216	Not Issued	030	04/25/2002	SIGNATURE SCHEMES BASED ON DISCRETE LOGARITHM WITH PARTIAL OR TOTAL MESSAGE RECOVERY	NACCACHE, DAVID
10130937	Not Issued	030	05/24/2002	A METHOD FOR ENCODING LONG MESSAGES FOR ELECTRONIC SIGNATURE SCHEMES BASED ON RSA	NACCACHE, DAVID
10130943	Not Issued	030	05/24/2002	PROTECTION AGAINST THE IMPROPER USE OF AN INSTRUCTION IN A MEMORY	NACCACHE, DAVID
10148022	Not Issued	030	05/24/2002	METHOD FOR ACCELERATED TRANSMISSION OF ELECTRONIC SIGNATURE	NACCACHE, DAVID
10257130	Not Issued	030	10/09/2002	METHOD FOR CALCULATING CRYPTOGRAPHIC KEY CHECK DATA	NACCACHE, DAVID

10257411	Not Issued	020	10/11/2002	COUNTERMEASURE METHOD IN A MICROCIRCUIT, MICCROCIRCUIT THEREFORE AND SMART CARD COMPRISING SAID MICROCIRCUIT	NACCACHE, DAVID
10311698	Not Issued	030	05/30/2003	ACCESS CONTROL TO DATA PROCESSING MEANS	NACCACHE, DAVID
10398732	Not Issued	030	04/09/2003	METHOD FOR PROTECTION AGAINST FRAUD IN A NETWORK BY ICON SELECTION	NACCACHE, DAVID
10467718	Not Issued	030	08/12/2003	METHOD FOR MULTIPLYING TWO BINARY NUMBERS	NACCACHE, DAVID
10467928	Not Issued	020	03/22/2004	IDENTIFICATION MODULE PROVIDED WITH A SECURE AUTHENTICATION CODE	NACCACHE, DAVID
10817453	Not Issued	020	04/05/2004	CRYPTOGRAPHIC SYSTEM COMPRISING AN ENCRYPTION AND DECRYPTION SYSTEM AND A KEY ESCROW SYSTEM, AND THE ASSOCIATED EQUIPMENT AND DEVICES	NACCACHE, DAVID
10859978	Not Issued	030	06/04/2004	BIOMETRIC IDENTIFICATION METHOD AND DEVICE ADAPTED TO VERIFICATION ON CHIP CARDS	NACCACHE, DAVID

Inventor Search Completed: No Records to Display.

Search Another: Inventor **Last Name** NACCACHE **First Name** DAVID Search

To go back use Back button on your browser toolbar.

Back to PALM | ASSIGNMENT | OASIS | Home page

Day : Monday
Date: 2/21/2005

PALM INTRANET

Time: 20:45:03

Inventor Name Search Result

Your Search was:

Last Name = ROUSSEAU
First Name = LUDOVIC

Application#	Patent#	Status	Date Filed	Title	Inventor Name
09763159	Not Issued	161	08/16/1999	METHOD AND DEVICE FOR AUTHENTICATING WITH SYMMETRICAL ALGORITHM	ROUSSEAU, LUDOVIC
09936174	Not Issued	030	12/06/2001	METHOD FOR MONITORING A PROGRAM FLOW	ROUSSEAU, LUDOVIC
10311698	Not Issued	030	05/30/2003	ACCESS CONTROL TO DATA PROCESSING MEANS	ROUSSEAU, LUDOVIC

Inventor Search Completed: No Records to Display.

Search Another: Inventor Last Name: ROUSSEAU First Name: LUDOVIC Search

To go back use Back button on your browser toolbar.

Back to PALM | ASSIGNMENT | OASIS | Home page

Day : Monday
Date: 2/21/2005

PALM INTRANET

Time: 20:45:16

Inventor Name Search Result

Your Search was:

Last Name = GIRARD
First Name = PIERRE

Application#	Patent#	Status	Date Filed	Title	Inventor Name
06088670	4325055	150	10/26/1979	ANALOG-TO-DIGITAL CONVERTER	GIRARD, PIERRE
06129656	4331713	150	03/12/1980	PROCESS AND APPARATUS FOR THE CONTINUOUS COATING OF A SHEET ARTICLE, PARTICULARLY A WEB OF PAPER OR PAPERBOARD	GIRARD, PIERRE
06323312	Not Issued	161	11/20/1981	PROCESS AND APPARATUS FOR THE CONTINUOUS COATING OF A SHEET ARTICLE PARTICULARLY A WEB OF PAPER OR PAPERBOARD	GIRARD, PIERRE
06680991	Not Issued	163	12/13/1984	PROCESS AND APPARATUS FOR THE CONTINUOUS COATING OF A SHEET ARTICLE, PARTICULARLY A WEB OF PAPER OR PAPERBOARD	GIRARD, PIERRE
07591378	5167226	150	10/01/1990	COMBINED CLAPPING AND VIBRATING DEVICE FOR EXPELLING RETAINED OBSTRUCTIVE SECRETIONS IN THE LUNGS	GIRARD, PIERRE
08683851	5827616	150	07/19/1996	COATED GREASEPROOF PAPER AND PROCESS FOR MANUFACTURING IT	GIRARD, PIERRE
08886144	6144406	150	06/30/1997	ELECTRONIC PANORAMIC CAMERA	GIRARD, PIERRE
08988064	5965091	150	12/10/1997	FILLED PAPER FOR GAS FILTRATION	GIRARD, PIERRE
08988066	6224768	150	12/10/1997	FILTER PAPER FOR LADEN LIQUIDS	GIRARD, PIERRE

08988067	5984110	150	12/10/1997	DEVICE FOR PURIFYING LIGHTLY LADEN WATER	GIRARD, PIERRE
09508316	Not Issued	041	05/26/2000	PAPER OR CARDBOARD WITH IMPROVED PRINTABILITY	GIRARD, PIERRE
09647650	Not Issued	094	10/03/2000	PHOTOCATALYTIC COMPOSITION	GIRARD, PIERRE
09743187	6503447	150	01/05/2001	METHOD FOR PURIFYING GASEOUS EFFLUENTS BY MEANS OF PHOTOCATALYSIS, INSTALLATION FOR CARRYING OUT SAID METHOD	GIRARD, PIERRE
09936174	Not Issued	030	12/06/2001	METHOD FOR MONITORING A PROGRAM FLOW	GIRARD, PIERRE
10181053	Not Issued	071	10/10/2002	METHOD FOR PROTECTING AGAINST THEFT OF A PIN NUMBER IN (A) MULTI-APPLICATION SMART CARD (S) AND CHIP CARD(S) IMPLEMENTING SAID METHOD	GIRARD, PIERRE
10181884	Not Issued	030	10/10/2002	METHOD FOR PROTECTING AGAINST THEFT THE AUTHENTICATING VALUE OF MULTIPLE APPLICATION SMART CARDS, SMART CARDS THEREFOR AND TERMINALS DESIGNED TO RECEIVE SAID CARDS	GIRARD, PIERRE
10276920	Not Issued	030	01/03/2003	METHOD FOR PROTECTION AGAINST FRAUDULENT MODIFICATION OF DATA SENT TO A SECURE ELECTRONIC MEDIUM	GIRARD, PIERRE
10296547	Not Issued	030	11/25/2002	MAKING SECURE DATA EXCHANGES BETWEEN CONTROLLERS	GIRARD, PIERRE
10311698	Not Issued	030	05/30/2003	ACCESS CONTROL TO DATA PROCESSING MEANS	GIRARD, PIERRE
10343112	Not Issued	030	01/28/2003	METHOD FOR MAKING SECURE A SESSION WITH DATA PROCESSING MEANS UNDER THE CONTROL OF SEVERAL ENTITIES	GIRARD, PIERRE

10467763	Not Issued	019	01/01/0001	DYNAMIC MANAGEMENT OF ACCESS RIGHTS LISTS IN A PORTABLE ELECTRONIC OBJECT	GIRARD, PIERRE
10484524	Not Issued	020	09/20/2004	METHOD FOR PROTECTING PERSONAL DATA READ IN A TERMINAL STATION BY A SERVER	GIRARD, PIERRE
10520434	Not Issued	019	01/01/0001	MAKING SECURE DOWNLOAD APPLICATION IN PARTICULAR IN A SMART CARD	GIRARD, PIERRE
10838023	Not Issued	094	05/03/2004	PHOTOCATALYTIC COMPOSITION	GIRARD, PIERRE
06141425	4353504	150	04/18/1980	HIGH PRESSURE SNOW GUN	GIRARDIN, PIERRE
06802043	4717072	150	11/26/1985	SEQUENTIAL VALVE DRAIN FOR SNOW GUN	GIRARDIN, PIERRE
07246667	4914923	150	08/05/1988	METHOD OF COVERING ARTIFICIAL ALPINE- OR NORDIC-SKIING TRACKS WITH SNOW AND MEANS FOR IMPLEMENTING THE METHOD	GIRARDIN, PIERRE
08501047	Not Issued	169	08/19/1995	IMPROVED SNOW GUN	GIRARDIN, PIERRE
08513850	Not Issued	169	09/12/1995	SUPPORT FOR A PULVERISATION APPARATUS FOR A MIXTURE OF WATER AND AIR UNDER PRESSURE	GIRARDIN, PIERRE
08525658	Not Issued	161	09/29/1995	SPRAYING NOZZLE AND DEVICE FOR SPRAYING A MIXTURE OF WATER AND USING SAID NOZZLE	GIRARDIN, PIERRE

Inventor Search Completed: No Records to Display.

Search Another: Inventor **Last Name** GIRARD **First Name** PIERRE Search

To go back use Back button on your browser toolbar.

Back to PALM | ASSIGNMENT | OASIS | Home page

Your search matched **1** of **1128145** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

configuration management<and>html<and>applet Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

1 **Java, the Web, and software development**
Yourdon, E.;
Computer , Volume: 29 , Issue: 8 , Aug. 1996
Pages:25 - 30

[Abstract] [PDF Full-Text (864 KB)] **IEEE JNL**

Results 1 - 20 of 200 Result page: 1 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 An efficient and lightweight embedded Web server for Web-based network element management

Hong-Taek Ju, Mi-Joung Choi, James W. Hong

September 2000 **International Journal of Network Management**, Volume 10 Issue 5

Full text available: pdf(428.26 KB) Additional Information: full citation, abstract, references, index terms

An Embedded Web Server (EWS) is a Web server which runs on an embedded system with limited computing resources to serve embedded Web documents to a Web browser. By embedding a Web server into a network device, it is possible to provide a Web‐based management user interface, which are user‐friendly, inexpensive, cross‐platform, and network‐ready. This article explores the topic of an efficient and lightweight embedded Web server for Web‐based netw ...

2 Versioning and fragmentation: Fine-grained, structured configuration management for web projects

Tien Nhut Nguyen, Ethan Vincent Munson, Cheng Thao

May 2004 **Proceedings of the 13th international conference on World Wide Web**

Full text available: pdf(698.15 KB) Additional Information: full citation, abstract, references, index terms

Researchers in Web engineering have regularly noted that existing Web application development environments provide little support for managing the evolution of Web applications. Key limitations of Web development environments include line-oriented change models that inadequately represent Web document semantics and in ability to model changes to link structure or the set of objects making up the Webapplication. Developers may find it difficult to grasp how theoverall structure of the Web applica ...

Keywords: software configuration management, version control, web engineering

3 Web-based personalization and management of interactive video

Rune Hjelsvold, Subu Vdaygiri, Yves Léauté

April 2001 **Proceedings of the tenth international conference on World Wide Web**

Full text available: pdf(611.20 KB) Additional Information: full citation, references, citings, index terms

Keywords: J2EE, SMIL, dynamic content generation, interactive video, media asset management, micro-payment, video personalization

4 Web and e-business application: Content management on server farm with layer-7 routing

Mon-Yen Luo, Chu-Sing Yang, Chun-Wei Tseng
March 2002 **Proceedings of the 2002 ACM symposium on Applied computing**

Full text available: pdf(540.88 KB) Additional Information: full citation, abstract, references, index terms

Service replication on a server farm is becoming increasingly widespread as the explosive growth of the Web is straining the architecture of many Internet sites. Layer-7 routing, routing packets based on requested content, has been recognized as a powerful approach to distribute workload among these server farms. However, little attention has been given to how to configure content-related knowledge into the layer-7 routing mechanisms. In addition, the used data structures for storing content-rel ...

5 Why web-based network monitoring? Leveraging the platform

Ron D. Jenkins
May 1999 **International Journal of Network Management**, Volume 9 Issue 3

Full text available: pdf(494.08 KB) Additional Information: full citation, abstract, index terms

The increasing use of network monitoring and the growth of the Internet and intranets are converging trends that make IP network infrastructures the logical means of delivering network monitoring, using browser -bused clients. Copyright © 1999 John Wiley & Sons, Ltd.

6 Current technological impediments to business-to-consumer electronic commerce

Gregory Rose, Huoy Khoo, Detmar W. Straub
June 1999 **Communications of the AIS**

Full text available: pdf(479.36 KB) Additional Information: full citation, references, citings

7 A load cluster management system using SNMP and web

Myung-Sup Kim, Mi-Joung Choi, James W. Hong
November 2002 **International Journal of Network Management**, Volume 12 Issue 6

Full text available: pdf(355.47 KB) Additional Information: full citation, abstract, references, index terms

Clustered servers for Internet service is a popular solution to cope with the explosive increase in client requests. The high probability of service failure in cluster servers make the cluster management system necessary to provide high availability and convenient administrator control. In this paper, we present the design and implementation of a load cluster management system (LCMS) based on SNMP and Web technology. Our LCMS implementation has been deployed on a commercial ultra-dense server.

8 Architecture and performance of server-directed transcoding

Björn Knutsson, Honghui Lu, Jeffrey Mogul, Bryan Hopkins
November 2003 **ACM Transactions on Internet Technology (TOIT)**, Volume 3 Issue 4

Full text available: pdf(927.92 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Proxy-based transcoding adapts Web content to be a better match for client capabilities (such as screen size and color depth) and last-hop bandwidths. Traditional transcoding breaks the end-to-end model of the Web, because the proxy does not know the semantics of the content. *Server-directed transcoding* preserves end-to-end semantics while supporting aggressive content transformations.We show how server-directed transcoding can be integrated into the HTTP protocol and into the implementat ...

Keywords: HTTP, proxy, transcode, web

9 Performance and scalability of EJB applications
Emmanuel Cecchet, Julie Marguerite, Willy Zwaenepoel
November 2002 **ACM SIGPLAN Notices , Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 37 Issue 11
Full text available: pdf(306.80 KB) Additional Information: full citation, abstract, references, citings

We investigate the combined effect of application implementation method, container design, and efficiency of communication layers on the performance scalability of J2EE application servers by detailed measurement and profiling of an auction site server.We have implemented five versions of the auction site. The first version uses stateless session beans, making only minimal use of the services provided by the Enterprise JavaBeans (EJB) container. Two versions use entity beans, one with container- ...

Keywords: EJB container design, communication optimization, performance, profiling, scalability

10 Enabling full service surrogates using the portable channel representation
Micah Beck, Terry Moore, Leif Abrahamsson, Christophe Achouiantz, Patrick Johansson
April 2001 **Proceedings of the tenth international conference on World Wide Web**
Full text available: pdf(282.92 KB) Additional Information: full citation, references, index terms

Keywords: content distribution, dynamic content, mirroring, portability, replication, surrogate, web server

11 Integrating open hypermedia systems with the World Wide Web
Kenneth M. Anderson
April 1997 **Proceedings of the eighth ACM conference on Hypertext**
Full text available: pdf(1.00 MB) Additional Information: full citation, references, citings, index terms

Keywords: Chimera, World Wide Web, intergration, open hypermedia systems

12 Educational environments: Resource management portal for laboratories using real devices on the Internet
Stefan Zimmerli, Marc-Alain Steinemann, Torsten Braun
July 2003 **ACM SIGCOMM Computer Communication Review**, Volume 33 Issue 3
Full text available: pdf(343.16 KB) Additional Information: full citation, abstract, references, index terms

Internet-based distance learning is slowly gaining new territories and substituting current teaching methodologies. However, distance learning not only consists of transferring documents to web pages, but also of developing new concepts, methods, and implementation architectures. This article presents concepts and implementation issues for an example remote hands-on networking laboratory. The described course gives access to real network hardware via the Internet. In particular, authentication, ...

Keywords: computer networks laboratory, distance learning, hands-on training, resource management

13 Maintaining page coherence for dynamic HTML pages
Antonio Si, Hong V. Leong, Stanley M. T. Yau
February 1998 **Proceedings of the 1998 ACM symposium on Applied Computing**
Full text available: pdf(957.58 KB) Additional Information: full citation, references, citings, index terms

Keywords: Internet/intranet, World Wide Web, database caching, digital libraries

14 Tools and approaches for developing data-intensive Web applications: a survey
Piero Fraternali
September 1999 **ACM Computing Surveys (CSUR)**, Volume 31 Issue 3
Full text available: pdf(524.80 KB) Additional Information: full citation, abstract, references, citings, index terms

The exponential growth and capillar diffusion of the Web are nurturing a novel generation of applications, characterized by a direct business-to-customer relationship. The development of such applications is a hybrid between traditional IS development and Hypermedia authoring, and challenges the existing tools and approaches for software production. This paper investigates the current situation of Web development tools, both in the commercial and research fields, by identifying and characte ...

Keywords: HTML, Intranet, WWW, application, development

15 Supporting industrial hyperwebs: lessons in scalability
Kenneth M. Anderson
May 1999 **Proceedings of the 21st international conference on Software engineering**
Full text available: pdf(1.47 MB) Additional Information: full citation, references, citings, index terms

Keywords: open hypermedia systems, scalability, software engineering

16 Session summaries from the 17th symposium on operating systems principle (SOSP'99)
Jay Lepreau, Eric Eide
April 2000 **ACM SIGOPS Operating Systems Review**, Volume 34 Issue 2
Full text available: pdf(3.15 MB) Additional Information: full citation, index terms

17 SNMP through WWW
Ching-Wun 'Bo' Tsai, Ruay-Shiung 'Bo' Chang
March 1998 **International Journal of Network Management**, Volume 8 Issue 2
Full text available: pdf(376.25 KB) Additional Information: full citation, abstract, references, citings, index terms

In this article we propose a bilingual agent to accept either SNMP or HTTP commands and design several HTML pages to facilitate the task of network management. For network

elements that support only SNMP, the bilingual agent can act as a proxy, so that the traditional SNMP agent can also be queried through the Web browser. © 1998 John Wiley & Sons, Ltd.

18 The state of the art in locally distributed Web-server systems

Valeria Cardellini, Emiliano Casalicchio, Michele Colajanni, Philip S. Yu

June 2002 **ACM Computing Surveys (CSUR)**, Volume 34 Issue 2

Full text available: pdf(1.41 MB) Additional Information: full citation, abstract, references, citings, index terms

The overall increase in traffic on the World Wide Web is augmenting user-perceived response times from popular Web sites, especially in conjunction with special events. System platforms that do not replicate information content cannot provide the needed scalability to handle large traffic volumes and to match rapid and dramatic changes in the number of clients. The need to improve the performance of Web-based services has produced a variety of novel content delivery architectures. This article w ...

Keywords: Client/server, World Wide Web, cluster-based architectures, dispatching algorithms, distributed systems, load balancing, routing mechanisms

19 Help design challenges in network computing

Ben Gelernter

September 1998 **Proceedings of the 16th annual international conference on Computer documentation**

Full text available: pdf(1.12 MB) Additional Information: full citation, references, index terms

Keywords: documentation, help, information architecture, network computing, network computing architecture, online help, thin clients, user assistance

20 Enhancing workflows by web technology

Wolfgang Gräther, Wolfgang Prinz, Sabine Kolvenbach

November 1997 **Proceedings of the international ACM SIGGROUP conference on Supporting group work : the integration challenge: the integration challenge**

Full text available: pdf(1.34 MB) Additional Information: full citation, references, citings, index terms

Keywords: HTML, Internet, electronic circulation folder, workflow

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT
Int. J. Network Mgmt 2000; 10:261–275

An efficient and lightweight embedded Web server for Web-based network element management

By Hong-Taek Ju,* Mi-Joung Choi and James W. Hong

An Embedded Web Server (EWS) is a Web server which runs on an embedded system with limited computing resources to serve embedded Web documents to a Web browser. By embedding a Web server into a network device, it is possible to provide a Web-based management user interface, which are user-friendly, inexpensive, cross-platform, and network-ready. This article explores the topic of an efficient and lightweight embedded Web server for Web-based network element management. Copyright © 2000 John Wiley & Sons, Ltd.

Introduction

As the World-Wide Web (or Web) continues to evolve, it is clear that its underlying technologies are useful for much more than just browsing the Web. Web browsers have become the *de facto* standard user interface for a variety of applications. This is because Web browsers can provide a GUI interface to various client/server applications without a client application. An increasing number of Web technologies can also be applied to network element management.

Web-based network element management gives an administrator the ability to configure and monitor network devices over the Internet using a Web browser. The most direct way to accomplish this is to embed a Web server into a network device and use that server to provide a Web-based management user interface constructed using HTML,[5] graphics and other features common to Web browsers.[4] Information is provided to the user by simply retrieving pages, and information is sent back to the device using forms that the user completes. Web-based management user interfaces (WebMUIs) through embedded Web servers have

Hong-Taek Ju received his BS degree in computer science from Korea Advanced Institute of Science and Technology (KAIST) in 1989 and MS degree in Computer Science and Engineering from Pohang University of Science and Technology (POSTECH) in 1991. From 1991 to 1997, he worked at DAEWOO Telecom. Currently, he is a PhD candidate in the Department of Computer Science and Engineering, POSTECH. His research interests include distributed processing and network management.

Mi-Joung Choi received her BS degree in computer science from Ewha Womans University. She is currently a graduate student in the Department of Computer Science and Engineering, POSTECH. Her research interests include Web-based network management and policy-based network management.

James W. Hong is an associate professor in the Department of Computer Science and Engineering, POSTECH, Pohang, Korea. He has been with POSTECH since May 1995. Prior to joining POSTECH, he was a research professor in the Department of Computer Science, University of Western Ontario, London, Canada. Dr Hong received BSc and MSc degrees from the University of Western Ontario in 1983 and 1985, respectively, and PhD degree from the University of Waterloo, Waterloo, Canada in 1991. He has been very active as a participant, program committee member and organizing committee member for IEEE CNOM sponsored symposiums such as NOMS, IM, DSOM and APNOMS. For the last several years, he has been working on various research projects on network and systems management, which utilize Web, Java and CORBA technologies. His research interests include network and systems management, distributed computing and traffic engineering and planning. He is a member of IEEE, KICS, KNOM and KISS.

*Correspondence to: Hong-Taek Ju, DPNM Laboratory, Department of Computer Science and Engineering, Pohang University of Science and Technology, San 31, Hyojadong, Namgu, Pohang, Korea.
Email: juht@postech.ac.kr*

many advantages: ubiquity, user-friendliness, low development cost and high maintainability.

Embedded Web Servers (EWSs)[1-3] have different requirements, such as low resource utility, high reliability, security and portability, for which general Web server technologies are unsuitable. Above all, due to resource scarcity in embedded systems it is important to make EWSs efficient and lightweight. There are also design issues such as HTTP[6,7] and embedded application interface. In embedded Web server usage, Java applets can play an important role for making embedded Web servers truly useful for management applications.

In this paper, we present our research to develop an efficient and lightweight EWS for Web-based network element management. We first propose the architecture of an embedded Web server that can provide a simple but powerful application interface for network element management. We then present the design and implementation of POS-EWS, an embedded Web server that we have developed for Web-based network element management. Finally, we present the results of POS-EWS's performance and EWS optimization methods for making an efficient and lightweight EWS. There are many commercial EWS products on the market for Web appliances, but our work is a good example of making an efficient EWS suitable for Web-based network element management.

The organization of the paper is as follows. In the second section we present an overview of EWSs, and describe the EWS-WebMUI and EWS requirements. In the next two sections we present the EWS design and implementation of our proposed EWS architecture, respectively. In the fifth section we evaluate POS-EWS's performance and explain our methods for optimizing POS-EWS. In the sixth section we briefly investigate the available offerings of EWS products focusing on their features and the approximate code size needed. In the final section we summarize our work and discuss possible future work.

Embedded Web Servers and Web-based Management User Interface

In this section, we briefly overview embedded Web servers, comparing them with general Web servers. Also, we describe the EWS-WebMUI and EWS requirements that we must consider during development.

—Embedded Web Server—

General Web servers, which were developed for general-purpose computers such as NT servers or Unix and Linux workstations, typically require megabytes of memory, a fast processor, a pre-emptive multitasking operating system, and other resources. A Web server can be embedded in a device to provide remote access to the device from a Web browser if the resource requirements of the Web server are reduced. The end result of this reduction is typically a portable set of code that can run on embedded systems with limited computing resources. The embedded system can be utilized to serve the embedded Web documents, including static and dynamic information about embedded systems, to Web browsers. This type of Web server is called an Embedded Web Server (EWS).[1-3]

EWSs are used to convey the state information of embedded systems, such as a system's working statistics, current configuration and operation results, to a Web browser. EWSs are also used to transfer user commands from a Web browser to an embedded system. The state information is extracted from an embedded system application and the control command is implemented through the embedded system application. In many instances, it is advisable for embedded Web software to be a lightweight version of Web software. For network devices, such as routers, switches and hubs, it is possible to place an EWS directly in the devices without additional hardware.

—EWS-WebMUI—

WebMUI and EWS-WebMUI—The rapid proliferation of Web-based management makes it clear that schemes using HTTP and standard Web browsers provide benefits to both users and developers. Most Web-based management applications provide an interface to the status reporting, configuration, and control features of managed objects. Several such Web management approaches have been proposed thus far. Sun Micro-systems

is pushing its Java Management eXtension (JMX)[8] and Microsoft, Compaq and Intel are touting Web-based Enterprise Management (WBEM).[9] However, both approaches are sufficiently complex that many small network devices would find it very difficult to implement them.

By embedding a Web server, Web documents and management applications into an embedded system, a Web-based Management User Interface (WebMUI) can be provided directly to system administrators (an EWS-WebMUI). Therefore, an EWS-WebMUI is the direct result of embedding a Web server, Web documents and management applications into an embedded system. The Web documents give a display form of management information, a collection of manageable data that is monitored or configured for managing an embedded system.

By embedding a Web server in a network device, the device can serve up Web documents to any Web browser.

Advantages of EWS-WebMUI—By embedding a Web server in a network device, the device can serve up Web documents to any Web browser. These Web documents become the GUI interface to the device. Consequently, few techniques need to be learned for management interface of the new device. Because Web documents can be displayed directly from files that may be edited with either ordinary text editors (for HTML) or specialized authoring tools, it is easy to quickly prototype the look and feel of a WebMUI. Alternatives can be explored and reviewed without ever actually embedding the interface into the system. If the mechanisms used to embed the interface are properly designed, changes made to the Web documents can be quickly imported to the embedded system with little or no change to the management application code. This translates into the potential for better, more useful interfaces in less development time.

EWS-WebMUIs also have the advantage of a platform independent graphical user interface. The SNMP[10] management scheme usually consists of an SNMP based Network Management System (NMS). Most NMSs give users the option of using a graphical interface based on MS-Windows or X-Window as opposed to the command line interface. Most NMS users demand specific platforms, such as OS, or computer hardware in order to install and execute the NMS. By contrast, an EWS-WebMUI does not demand any specific platform because Web browsers are available for virtually all computers.

While the EWS-WebMUI concept appears straightforward and perhaps even commonplace, the implications are deeper than first appears. By placing the GUI within the device itself, the device is now self-contained and need not be matched with a corresponding version of a user management application program; the problems inherent in providing separate user interface software disappears; there is no risk of the user having an old version of the user application software that does not support all the features of latest devices; and users can upgrade some systems to the latest release without having to change the management software they use because the necessary part of upgrade is only the EWS-WebMUI. Consequently, there are no porting or distribution efforts for the user application program.

Additionally, it is usually possible to upload Web documents to the embedded system so that a device can receive an upgrade to its management interface from a remote location on the network. This feature makes it possible for developers to upgrade all devices over the network from the one point. High maintainability for EWS-WebMUI is a direct result of ease of Web document development and one point upgrade.

Design

In this section, we present our design result that includes a functional architecture and a process structure of EWS.

—EWS Architecture—

We have designed an EWS that consists of five parts: an HTTP engine, an application interface module, a virtual file system, a configuration module, and a security module. The design architecture of our EWS is illustrated in Figure 1.

The most important part of the EWS is an HTTP engine, which serves a client's request. The

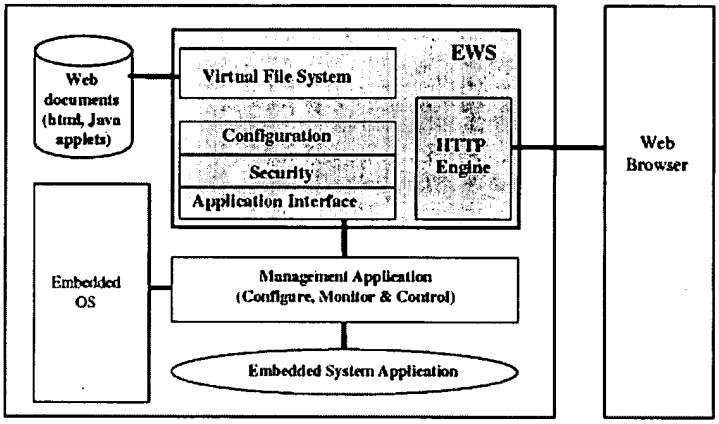


Figure 1. EWS architecture

minimum requirement for an HTTP engine is that it must be compliant with HTTP specifications. Unlike general Web servers that start a new thread or process whenever a new connection is made, normally an HTTP engine supports multiple simultaneous users while running as a single process. The number of processes that the server requires can impact on both RAM usage, due to the stack space per task, and CPU usage. Next, we explain an HTTP transaction process using a state transition diagram.

In an EWS, the application interface module enables developers to add new management functionality. With any off-the-shelf Web authoring tool, it can merge Web documents with management application programs to generate specific dynamic management information. This module provides mechanisms for interacting with the embedded application. Embedded Web server software must provide mechanisms for the embedded application to generate and serve Web pages to the browser, and to process HTML form data submitted by the browser. One possible solution is modeled after the Common Gateway Interface (CGI)[15] found in many traditional Web servers. In this model, each URL[16] is mapped to a CGI script that generates the Web page. In a typical embedded system, the script would actually be implemented by a function call to the embedded application. The application could then send raw HTML or other types of data to the browser by using an interface provided by the embedded Web server software.

Another solution is to use Server-Side Include (SSI).[5] With this approach, Web pages are first developed and prototyped using conventional Web authoring tools and browsers. Next, proprietary markup tags that define server-side scripts are inserted into the Web pages. The marked-up Web pages are then stored in the device. When a marked-up Web page is served, the embedded Web server interprets and executes the script to interface with the embedded application. In order to offload substantial Web server processing from the embedded system at run time, a preprocessor tool can be used. The preprocessor enables sophisticated dynamic Web-page capabilities by performing complex tasks up front and generating an efficient and tightly integrated representation of the Web pages and interfaces in the embedded system.

The virtual file system (VFS) provides the EWS with virtual file services, which are *file_open* for opening the file, *file_read* for reading the file, and *file_close* for closing the file after reading. The file system has a data structure storing file information such as file size, last modified date, etc. The data structure for an HTML documents file needing dynamic information must store the pointer of the script and the function name called by the script. To construct this VFS we need a Web compiler. The Web compiler supports any format, such as Java, GIF, JPEG, PDF, TIFF, HTML, text, etc. It compiles these files into intermediate C-codes and then compiles & links them with the Web Server codes. The resulting structure does not require a

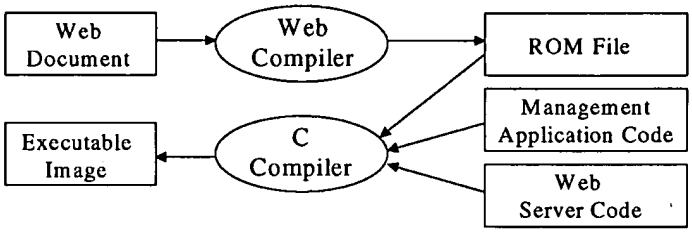


Figure 2. Process of a web server making a virtual file system

file system, yet the files are organized like in a file system—a virtual file system. The Web browser traverses this virtual file system just as if it were an actual file system. Figure 2 illustrates the process of a Web server making a virtual file system.

Security is an important concern in network management. Therefore, an EWS generally has a security and/or configuration module. Security is accomplished by defining security realms on a server and username/password access to each realm. When a request comes in for an object in a protected realm, the server responds with a response code of 401 (Unauthorized). This will force a browser to prompt the user for a username/password pair. The original object request will be resubmitted with the username/password, base-64 encoded, in the request header. If the server finds the login correct, then it will return the requested object, otherwise, a 403 forbidden response is returned. The configuration module provides the administrator with the functionality to set the embedded Web server configuration from any standard Web browser. The configuration environment variables passed at startup define the number of concurrent connections, socket port, own host name, root file path, default 'index', inactivity timeout and time zone. Common usage of Web browsers makes it a more important matter to protect abnormal access to the sensitive information of network devices, especially those that involve equipment configuration or administration.

—EWS Process Structure—

We designed an EWS as a finite state machine (FSM), which processes an HTTP request in a sequence of discrete steps. Figure 3 shows the state transition diagram of the HTTP engine. In order to support multiple connections in a single thread environment, multiple finite state machines are run by a scheduling system which uses a lightweight

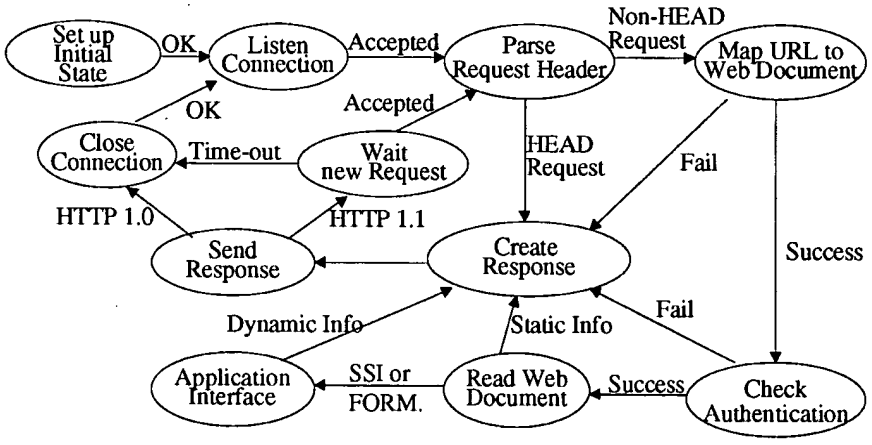


Figure 3. EWS finite state machine

task structure. It consists of a pointer to the function being run, a variable holding the state in the FSM, and a flag indicating whether the FSM can be run or blocked. The scheduling system allocates an available FSM for an accepted connection, checks each FSM to see if it is blocked or runable and, if it is runable, moves the FSM one step.

Each state in an FSM can check for the presence of data that is ready to be processed at the entry point; if none is ready, the FSM can block itself until data arrives. When data becomes available at the entry point, the FSM can then be unblocked so its handler can perform the task of state, and turn over the result to the next state by changing the state flag and pointer to the handler.

The following list describes the behavior of each state.

- *Set up Initial State*: Set up the task structure for an FSM. The task of this state is performed at the server initial time for all FSMs.
- *Listen Connection*: Check to see if any request is allocated to this FSM.
- *Parse Request Header*: Read the HTTP message, parse the HTTP header and store the parsing result.
- *Map URL to Web Document*: Determine type of application interface and store a pointer to the handler.
- *Check Authentication*: Force authentication of the user upon the URL and user name/password.
- *Read Web Document*: Read Web document from virtual file system.
- *Application Interface*: Call application function upon the URL.
- *Create Response*: Create HTTP response header.
- *Send Response*: Send HTTP header and Web document.
- *Wait new Request*: Wait for a new HTTP request from the same TCP connection if the received request says HTTP/1.1 support.
- *Close connection*: Close the TCP connection.

—EWS Extended Architecture for EWS WebMUI—

As mentioned earlier, only the scheme of HTTP and HTML is client-driven. One side effect is that once a page is served to the Web browser it becomes static: it does not change even if management data has been altered on the server side. For a user seeking a device, which is dynamic, this is not very appealing.

To be useful for management applications, pages must be constructed dynamically so that real-time data can be placed alongside static HTML in the same page. For common types of real-time data, such as traffic monitoring and CPU load, users want to see data displayed in a dynamic graphic form. This is where Java applets[17] and/or CORBA[18,19] objects come in. Java applets are automatically downloaded by a browser as separate applications that get used within an HTML page. Once the applet is loaded, it has control over where it gets its data and how to display or manipulate that data. Java applets by nature are cross-platform and will act the same within any browser.

Simple Network Management Protocol (SNMP)[10,20] is the most widely used management framework for managing network devices on the Internet. Its protocol is simple enough that it can be implemented in small platforms without much difficulty. Now most network devices are equipped with an SNMP agent. With integration of SNMP and the EWS-WebMUI, the advantages of EWS-WebMUI are preserved without the giving up the SNMP implementations.

The EWS extended architecture gives an integration platform. Figure 4 illustrates the EWS extended architecture for an EWS-WebMUI. The ultimate solutions is to make the EWS-WebMUI a user interface to communicate with the network device via SNMP. Java implementation of SNMP mediates between an SNMP agent and a Web browser. The Java SNMP source code is written and compiled to produce a Java SNMP applet. This applet is stored in a network device and is transferred by the EWS to the browser over the network at run time. After loading on the JVM of a browser, the Java SNMP applet communicates with the SNMP agent in the network device and enables the administrator to control and monitor the network device through the browser, using SNMP messages. In addition to the Java SNMP applet, the network device in this scenario must store at least one HTML document containing reference to the applet. The HTML document is loaded into the Web browser and then the Web browser would automatically request the Java SNMP applet referenced by the previous HTML.

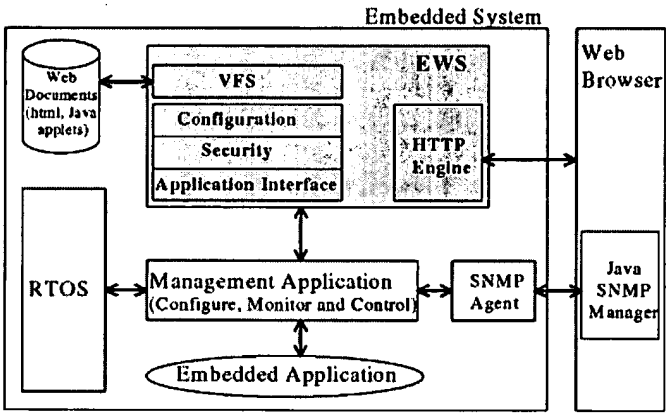


Figure 4. EWS extended architecture for WebMUI

The code size of a Java SNMP applet is small enough to be embedded into the network device because SNMP is simple (three basic massage types and a simple message format) and light protocol (uses UDP as its transport protocol, and thus does not have connection setup and acknowledgement overhead). SNMP defines an alert message and traps, which can be directed toward one or more trap receiver stations. If a trap management application is implemented as the Java SNMP applet and loaded from the network device, traps can be collected and viewed together by the Java SNMP applet where appropriate responses will follow.

Implementation

We have implemented an HTTP/1.1 compliant embedded Web server based on the EWS design presented in the previous section. We call this system POS-EWS, which stands for POStech-Embedded Web Server. To demonstrate how POS-EWS works, we have applied our POS-EWS to the element management of a commercial Internet router. The C programming language, commonly used in an embedded system, is used throughout the server implementation. We have implemented POS-EWS on the Xinu OS using the MPC 860 processor.

—Features of POS-EWS—

POS-EWS implements a subset of the HTTP features typically required for use in an embedded system. To reduce the TCP connection resources, HTTP/1.1 permits a persistent TCP connection to be established for as long as the Web browser requires access to the server. For providing up-to-date dynamic information, the server needs to control the cache mechanism that is also included in HTTP/1.1. The cache control and persistent TCP connection is essential for an EWS, and POS-EWS supports these two features.

In an embedded software system, dedicating a unique process or thread to every incoming connection is usually impractical due to the memory overhead required and, in some cases, to the lack of embedded OS support for multiple processes. When developing POS-EWS, we approached the problem of supporting multiple connections in the context of a single thread by implementing a finite state machine, which processes a request as a sequence of discrete steps. With multiple finite machines in a single thread, several connections can be activated at once, where each state machine representing a specific connection is scheduled to process in a round-robin manner. POS-EWS imposes a deterministic scheduler for handling multiple finite state machines.

For an embedded system, which may not need the full features of a file system, POS-EWS uses a Virtual File System (VFS) which can provide a limited set of read-only files built into the ROM. The VFS can be used with or without a real file system. If a real file system exists, the VFS will forward the file request to it. Using the VFS generator, which is one component of the

POS-EWS preprocessor compiler, the compressed HTML file for use by the EWS is created. The file will be decompressed by the VFS prior to use by POS-EWS.

POS-EWS supports the SSI style application interface. A proprietary tag can be included in a Web page so that when the page is requested it will cause POS-EWS to execute the function specified in the Web page using the tag. The function returns string data directly to POS-EWS to be used as part of the requested Web document. This allows the inclusion of dynamic management data directly into a loading HTML document, such as the current time or communication port status. We implemented this interface style via a table of name and pointer to functions. The table is constructed from the POS-EWS preprocessing compiler using the construction method explained below. Another application interface method is the FORM processing interface method.[5] The HTML FORM keyword allows the browser to send input back to the server by issuing a POST HTTP message. This feature is useful if there are control commands or configuration settings that need to be sent to management applications. Upon receipt of a POST message, POS-EWS calls a function that parses input from the browser and performs an action based on what it found in the input. Like the SSI style interface, this type of interface is also implemented by a table and preprocessing.

POS-EWS also supports state management using HTTP cookies.[21] A cookie is a record that contains management data for a manager to set. It is stored on Web browsers, and is sent to Web servers each time a manager sends a request to a Web server. Cookies are useful for having a Web browser remember some specific information which the Web server can later retrieve. A server, when returning an HTTP object to a client, may also send a piece of state information which the client will store. This simple mechanism can be used in management applications.

—POS-EWS Web Compiler—

We have also developed a Web compiler[22] for constructing a virtual file system (VFS) and efficient SSI application interface. Interpreting scripts at run time results in full scanning for the HTML file, which may impact system performance. The Web compiler can offload POS-EWS's scanning results by recording the position of a tag with the HTML file in the VFS. The server reads the HTML file before the starting point of a tag, calls the script function and proceeds with reading the HTML.

An example of an HTML and a subset of a compilation result are shown in Figure 5. In this example, the content of *sysname.html* is converted into a character array by the name of *sysname_html*, which is the result of simple conversion from file name to C language array name, i.e., changing the dot to an under-score.

The structure *vf* is a container for storing file information such as file size, last modified date, etc. The pointer value of the converted character

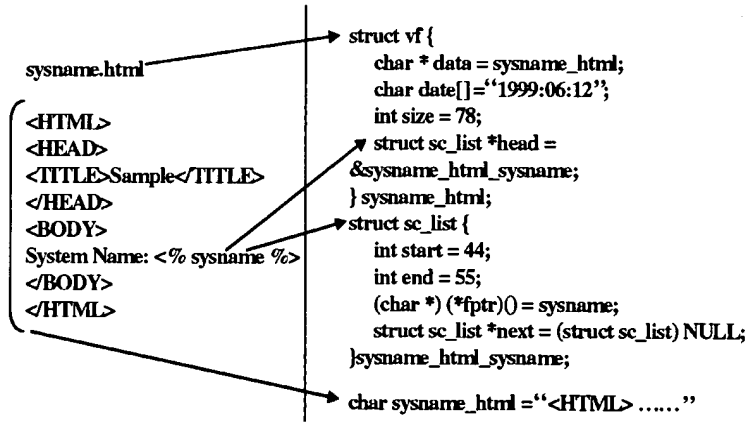


Figure 5. Virtual file system code

array, *data*, is one of the most important elements in the structure because this value is used to read real content by POS-EWS at run time. The structure *sc_list* is used to make a linked list for script functions. The header pointer for the linked list is also one element in the *vf* structure. POS-EWS uses this pointer value for calling the script function. The structure *vf* has an additional variable for supporting the file interface functions, for example, file read pointer, file state flag, etc. With the file interface functions such as file open (*vf_open*), file read (*vf_read*) and file close (*vf_close*), generated C codes become a complete virtual file.

Optionally, the Web compiler can also compress the Web documents. HTML is easily compressed as much as 50% with almost no run time memory required for decompression. HTTP/1.1 supports compressed file transfer from the Web server to Web browser. The Web document is stored in compressed form, transmitted directly, and decompressed by the Web browser. HTTP/1.1 can convey the information of compressed documents in the *Accept-Encoding* and *Content-Encoding* header fields. More importantly, it indicates what decompression algorithm will be required to remove the compression encoding. The following algorithms, as well as others, are registered in standard HTTP/1.1: *gzip* (generated by the GNU gzip program), and *compress* (produced by the common UNIX file compression program *compress*). Because the algorithms minimize the ROM space used, storing a reasonable size of Web documents on the device has a negligible impact on embedded system resources. For POS-EWS, we have used the *gzip* algorithm to compress at preprocessing and decompress at run time.

The results of implementation can be summarized as follows: the POS-EWS Web compiler converts Web documents to be stored in the virtual file system to compressed C arrays as a virtual file. Then it creates a directory data structure in order to store the file information in the virtual file system. Library functions for the file interface are supported without any RTOS dependency.

—POS-EWS Management Application Example—

Management information can be classified by the update period, direction of information flow or object of information source. From the viewpoint of update period, some management information changes dynamically, and some does not change at all. Furthermore, some information possesses real-time characteristics. Regarding the direction of information flow, some information can originate from a Web browser and go to a Web server and vice-versa.

As shown in Figure 6, there are four methods to retrieve data from an embedded system using POS-EWS. Method (a) is the most basic method to display data in a Web browser. POS-EWS reads data from a file and sends it to the browser. It

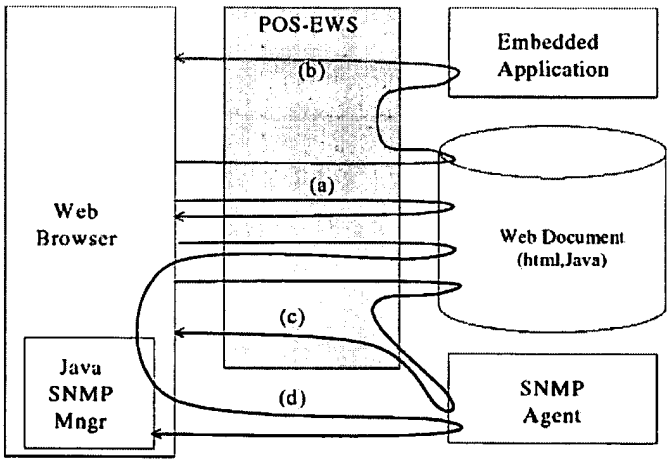


Figure 6. POS-EWS WebMUI mechanism

is suitable for static information like menu, image and so on. Method (b) is the second method, where POS-EWS reads the requested HTML file, calls an embedded application function in accordance with the script tags, replaces the tags of HTML with the result of the application function all format, and sends it to the browser. This method is suitable for showing dynamic information of the system. Circle (b) in Figure 7 shows the user-selectable port name retrieved by this method.

Method (c) is the same as (b) except for the information provider. POS-EWS retrieves an SNMP MIB value instead of an application function return value in replacing tags in HTML. It is suitable if management information is defined in SNMP MIB. It has the advantage of showing static and dynamic information of a system using an SNMP MIB database without additional SNMP traffic.

In method (d), POS-EWS sends the Java SNMP manager applets to the Web browser when requested, and the browser executes them. The Java SNMP manager continuously sends SNMP GET messages to the SNMP agent in the system for displaying real-time data. This method is suitable for showing real-time changes of system status and SNMP Trap information. Circle (d) in Figure 7 is Java applet which displays status of each port in real-time.

For validation of our work on the design and implementation of an efficient and lightweight EWS, we have used our POS-EWS for the network element management of a commercial Internet router. Figure 7 shows the display result of the WebMUI. Circles (a), (b), (c), (d) in Figure 7 show four different ways to retrieve data from an embedded system using the POS-EWS mechanism explained in Figure 6.

Performance Evaluation

We developed POS-EWS for Web-based network element management. In this section, we evaluate POS-EWS's performance in areas such as code size, run-time memory, CPU usage and connection capability. We also explain the methods used in optimizing our POS-EWS.

—Performance Metrics—

The performance of a Web server is dependent upon a number of variables: the server hardware and operating environment, the server application,

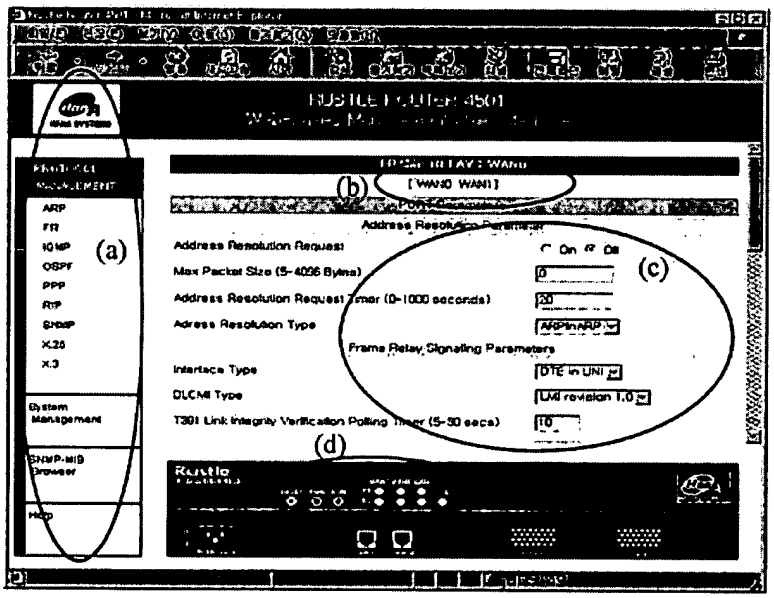


Figure 7. POS-EWS application interface example

The performance of a Web server is dependent upon a number of variables: the server hardware and operating environment, the server application, the network protocol, and the network, hardware, bandwidth and traffic load.

the network protocol, and the network, hardware, bandwidth and traffic load. Perception of that performance depends also on variables on the client side: the client platform and operating environment, and the Web client.[23] There are four metrics most often used to measure the capacity of general Web servers.

- Requests per second (rps or HTTP ops/sec), which is connections served or requests made per second.
- Throughput in bytes per second, which is dependent upon the bandwidth of the data pipe.
- Round-trip time (RTT), which is a measure of how long it takes for a packet to be sent from the client plus the time a response from the server takes to be received by the client, completing the request.
- Error rate, which is a measure of how many HTTP requests were lost or not handled by a server.

The two key elements of HTTP performance are latency and throughput.

- Latency is measured by the RTT and is independent of the object size.
- Throughput is a measure of how long it takes to send data, up to the carrying capacity of the data pipe. Improving throughput is simply a matter of employing faster, higher bandwidth networks.

Basically, the performance of a general Web server can be measured by timing how long a server takes to respond to a request and to count the number of bytes delivered per unit time. But EWSs have different performance metrics from general Web servers.

In general, embedded systems must minimize requirements for system resources such as CPU and memory: the embedded Web server is acting as a secondary feature of the system and should avoid interfering with the system's main purpose as much as possible. The most important task of an embedded system is to perform mission-critical and real-time applications. An EWS is often the lowest priority service in the system, so end-users can wait hundreds of milliseconds for a response (an eternity compared to the low-latency requirements of many embedded real-time applications). Therefore, RTT and throughput are not important metrics for an EWS.

The performance element of POS-EWS are code size (memory footprint), run-time memory, the CPU usage and maximum user connectivity (the capacity of POS-EWS). The code size is approximately 30 Kbytes, the average run-time memory is 64 Kbytes, with the HTTP server having the lowest priority. The POS-EWS supports multiple, simultaneous HTTP transactions and multiple users. By evaluating the system performance we can determine how much an EWS impacts its embedded system. The code size of our POS-EWS is very small so that it puts little impact on the resource scarcity problems.

—POS-EWS Optimization—

We implemented POS-EWS as a finite state machine (FSM) to improve its performance. We approached the problem of supporting multiple connections in the context of a single process and thread by implementing an FSM, which processes an HTTP request as a sequence of discrete steps. The 'process-per-connection' architecture, which forms a new process for each connection, while it goes by the stateless model of the HTTP scheme, is less than efficient. The time and resources required by the fork and exec operations are significant, particularly in light of the fact that a typical Web request is very short.[24] But the FSM supporting single thread is run by a small scheduling system that uses lightweight task structures. This makes the CPU usage and the memory footprint reasonable.

We also give our HTTP engine more improved performance following the HTTP standard. We implemented POS-EWS to keep TCP connections open and reuse them by the Keep-Alive option. Therefore, the cost of opening a new connection for each transaction can be eliminated by reusing

existing TCP connections. In this way the transaction time will be the time to send a request plus the RTT plus the processing time on the server plus the time to send the response. When the Web server keeps TCP connections open and does not close them at end of an HTTP exchange, the maximum number of available TCP connections will be reached. The Web server then closes the oldest idle connection first. HTTP version 1.1 specifies the optional use of the Keep-Alive connection.[7] Requesting a Keep-Alive connection when GETing a file means the browser can reuse the connection to get subsequent files from the server.

The POS-EWS Web compiler preprocesses and compresses Web pages and images into compilable ANSI-C code. This allows pages to be developed using standard HTML tools, and stored internally in an efficient format. The Web compiler makes it possible to minimize the application memory footprints through intelligent compression. Shared and nested pointer techniques are used for additional memory savings. The Web compiler reduces the processing time through preprocessing.

We used APIs to extend a server's functionality. Instead of having to parse the incoming amorphous stream of form data, POS-EWS uses options for receiving it, often preformatted and type converted into C struct fields, ready for program use. Using an API instead of a CGI has the advantage of integrating the extensions to the server within the server process.[25] This eliminates the need to go to the OS for communications between the server and the script. Such a C level interface can save much time and simplify CGI programming immensely.

Related Work

In this section, we briefly investigate embedded Web server products, focusing on their features. Web servers can have a range of capabilities and still be http-compatible. A number of commercial EWS products have appeared on the market, each with its own particular value position. There are also many computer communities exploring small Web servers. They make a Web server the size of a matchbox. The Matchbox Server[35] is a single-board computer with a 16 MB RAM, and 16 MB flash ROM, big enough to hold a useful amount of Linux including the HTTP daemon. In addition, the iPic Web server[36] is the smallest Web server. It is about the size of a mere match-head. The single chip computer runs the iPic web-server, the world's tiniest implementation of a TCP/IP stack and an HTTP web-server.

Table 1 compares the features of a number of commercially available Embedded Web Server implementations and our POS-EWS. Blank columns represent features not supported or we could not find appropriate information on them from the available literature. We summarize the offerings available and the approximate code size needed. This range does not necessarily reflect differences in code efficiency, however. Most EWSs offer small footprints lower than 100 Kbytes for low resource utility, dynamic content generation of SSI type mechanism, some kind of page compression, and options for security/authentication and porting layers to accommodate custom file system and TCP/IP stack.

All of the above products work with any standard browser, including Internet Explorer, Netscape Navigator or Communicator, Hot Java and Mosaic, on all platforms, allowing any network-connected browser user to easily access any device. As well, all serve multiple, simultaneous users. Processing multiple requests may be important if more than one user is to access the embedded system at the same time, or if the system is to report itself busy to potential users. A few support a Web compiler and Virtual File System (VFS), which are essential features for enhancing Web ability and efficiency. Only RomPager[27] and our POS-EWS clearly specify HTTP cookies for state management. POS-EWS has full features of EWS to support the functionality of EWS.

Also, our POS-EWS provides effective integration mechanisms into embedded management applications. POS-EWS offers four basic interface mechanisms for use between a management application and application of an embedded system and an embedded Web server or Web documents. A developer can choose an appropriate interface mechanism depending on the characteristics of management information or types of Web documents. We testified them through applying POS-EWS to management of a commercial router. The integration mechanisms into embedded management applications are important when an EWS is applied to network element management.

Company & Product	OS supported	CPU supported	HTTP code size (version)	Features					
				SSI	VFS	Compiler	Compression	Security (encode)	Cookie
Agranat Systems, EmWeb[26]	No OS	Any CPU with a C compiler	25 kbytes (1.1)	O	O	O	Proprietary	Basic + Digest	
AllegroSoft, RomPager[27]	Any RTOS, No OS	Any CPU with a C compiler	10–40 kbytes (1.1)	O	File Sys.	O	Dictionary	Basic	O
BVM IntraScada, Web Server[28]	OS-9	CPU32	<100 kbytes (1.1)						
ccelerated Technology, Nucleus Embedded Software[29]	Nucleus Plus	x86,68K,ARM, 683xx,SPARC, PowerPC,SH, H8/300H, TMS320C3x, MIPS,4x/5x/2x, Panasonic MN10200	40 kbytes (1.0)	O			Propritary	Basic (DES)	
Spyglass, MicroServer[30]	LynxOS, QNX, pSOS, OS-9, VxWorks	Any CPU with a C compiler	35–110 kbytes (1.1)	O			None	Baisc + Digest + SSL	
QNX Software Systems Ltd, QNX Internet Toolkit[31]	QNX real-time OS	x86, Pentium Pro, AMD Elan	106 kbytes (1.1)	O				Basic + Digest (encode)	
Magma, Lava[32]	Any RTOS	Any CPU with a C compiler	15–40 kbytes (1.0)	O			Proprietary	Basic + SSL	
Quiotix, QEWS[33]	pSOS, LynxOS, VxWorks	Any CPU with a C compiler	45–50 kbytes (1.0)	O	O		GZIP	Basic	
Web Device, Pico Server[34]	LynxOS, Nucleus Plus, pSOS, VxWorks	Any CPU with a C compiler	15–30 kbytes (1.0)	O	O		ZIP-like	Basic + Digest + SSL	
POSTECH, POS-EWS	Real-time Xinu, pSOS	Any CPU with a C compiler	30 kbytes (1.1)	O	O	O	GZIP/CSS-Style	Basic + Digest (base64)	O

Table 1. Comparison of EWS products

Conclusion and Future Work

Web servers are already being built into many network devices today. In the near future, we can expect this trend to grow even further to home appliances, medical instruments and industrial equipments. Embedded Web servers for Web-based network element management provides an administrator with a simple but enhanced and more powerful user interface without additional hardware. Although without the management application specific aspect of embedded Web server technology these advantages may disappear.

In this paper, basic technical concepts on efficient and lightweight embedded Web servers were described and technical issues for their management application interface to network devices were explored. Also, we presented our design and implementation of an HTTP/1.1 compliant embedded Web server (called POS-EWS) based on the proposed architecture. The EWS-WebMUI architecture provides an easy integration platform with an SNMP agent. That is, POS-EWS provides easy and effective integration mechanisms for embedded management applications. This 'web-ification' of network devices has generated a new philosophy for network element management.

We also ported POS-EWS to the pSOS OS using MPC 860 processor. We plan to optimize our POS-EWS even further and make it more powerful in its application to home appliances, office equipment, and industrial products. We must check the reliability of our POS-EWS for porting to other embedded systems. This is important because POS-EWS is a part of embedded systems requiring reliability. We plan to measure quantitative performance data of POS-EWS. We also plan to port POS-EWS to other CPUs and embedded OSs. Future work involves investigating methods for network management of devices equipped with EWSs.

References

1. McCombie B. Embedded web servers now and in the future. *Real-Time Magazine* 1998; No. 1, 82–83, March.
2. Wilson A. The challenge of embedded internet. *Electronic Product Design* 1998; 31–2, 34, January.
3. Agranat I. Embedded web servers in network devices. *Communication Systems Design* 1998; 30–36, March.
4. Wellens C, Auerbach KK. Towards useful management. *The Simple Times* 1996; **4**(3): 1–6, July.
5. W3C. *HTML 4.0 Specification*. Internet Draft REC-html40-19980424, HTML Working Group, April 1998.
6. W3C. *Hypertext Transfer Protocol-HTTP/1.1*. Internet Draft draft-ietf-http-v11-spec-rev-06, HTTP Working Group, November 1999.
7. Fielding R. *Hypertext Transfer Protocol—HTTP/1.0*. RFC 1945, IETF HTTP WG, May 1996.
8. Sun Microsystems Inc. *Java Management API Architecture*, June 1996.
9. WBEM Consortium. Web-Based Enterprise Management Proposal. *HyperMedia Management Protocol Overview*. Revision 0.04. July 1996.
10. Case J, Fedor M, Schoffstall M, Davin C. The simple network management protocol (SNMP). RFC 1157, May; Mullaney P. Overview of a web-based agent. *The Simple Times* 1990; **4**: No. 3, July, 1996.
11. Berners-Lee T. Uniform Resource Locators (URL). Internet Draft, March 1994.
12. Fielding R. Hypertext Transfer Protocol—HTTP/1.1. RFC 2068 IETF HTTP WG, August 1996.
13. Touch J, Heidemann J, Obraczka K. *Analysis of HTTP Performance*. USC/Information Sciences Institute, June 1996.
14. Frystyk Nielsen H. *et al.*, Network Performance Effects of HTTP/1.1, CSS1, and PNG. W3C, June 1997.
15. CGI, *http://www.w3c.org/cgi/*.
16. Berners-Lee T. Uniform Resource Locators (URL). Internet Draft, March 1994.
17. Arnold K, Gosling J. *The Java Programming Language*. Addison-Wesley: Reading, MA, 1996.
18. OMG. *The Common Object Request Broker: Architecture and Specification, Revision 2.0*, July 1995, OMG TC Document.
19. Kong JY, Hong JW. *A CORBA-based Management Framework for Distributed Multimedia Services and Applications*. Technical Report PIRL-TR-97-1, POSTECH, Korea, March 1997.
20. Tsai CW, Chang RS. SNMP through WWW. *International Journal of Network Management* 1998; **8**: 104–119.
21. Kristol D, Montulli L. HTTP State Management Mechanism. RFC 2109, February 1997.
22. Ju HT. *POS-EWS Web Compiler for Supporting an Effective Application Interface*. PIRL Technical Report, POSTECH, PIRL-TR-99-3, December 1999.
23. Rubarth-Lay J. Keeping the 400lb. gorilla at Bay. *Optimizing Web Performance* May 1996.
24. Padmanabhan V, Mogul J. Improving HTTP latency. *Computer Networks and ISDN Systems*, December 1995; **28**: 25–35.
25. Edwards N. Rees O. *Performance of HTTP and CGI*. ASNA.

26. Agranat Systems. EmWeb. *http://www.emweb.com/.*
27. AllegroSoft. RomPager. *http://www.allegrosoft.com/.*
28. BVM IntraScada. Web Server. *http://www.bvmltd.co.uk/.*
29. Accelerated Technology. Nucleus Embedded Software. *http://www.atinucleus.com/.*
30. Spyglass. MicroServer. *http://www.spyglass.com/.*
31. QNX Software Systems Ltd. QNX Internet Toolkit. *http://www.qnx.com/.*
32. Magma Information Technologies. LAVA. *http://www.magmainfo.com/.*
33. Quiotix Corporation. QEWS. *http://www.quiotix.com/.*
34. Web Device Inc. Pico Server. *http://www.webdevice.com/.*
35. Stanford University. Matchbox Server. *http://wearables.stanford.edu/.*
36. University of Massachusetts. iPic Web server. *http://www-ccs.cs.umass.edu/~shri/iPic.html/.* ■

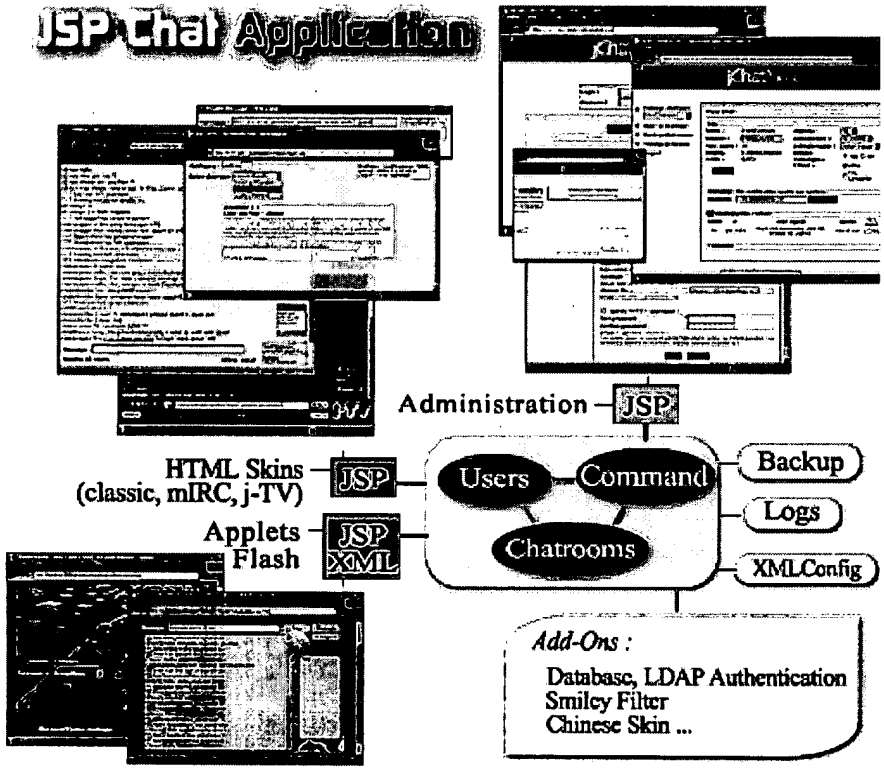
Applets
Servlets
Projects
Services
Links
About

jChatBox support

Here is **jChatBox 2.6**, the JSP™ **Chat Application**. It allows to op manage multilanguage chatrooms. System user can open and contr multiples chatrooms. Moderators can manage users (list, ban, kickoff), r blacklist, generate transcripts, do backup, and apply chatrooms parame such as max users, language, filters, performance, ... Server side needs Servlets/JSP. Client side could be HTML/CSS/JavaScript, Applet, Flash o Application such as SWING. Chatrooms are easily customizables thanks skins. They can be extended by designers or programmers through the jChatBox API. jChatBox is ready for XML communication thanks to its XI Connector service. An **Add-Ons** section is also available with tools like s filter, **web cam** skin, **database** and **LDAP** authentication, **chinese** and **russian** translations, **Flash** client, **self-closing** empty chatrooms, XMLTranscriptor, **Load benchmark ...**

Any Question ?
Try our Forums !

Translations !
[Portuguese]
[Russian]
[German]



Non-Technical description **Technical description** **Do**

Documentation & Guides **OnLine Demontration** **Ad**

last update : 01/07/2004
You can now license the source code of jChatBox 2.6 here !

Non-Technical description :

Here is the non-technical description to let you know about what jChatBox does but not how. Let's go for jChatBox features !

Non-Technical : Chat Server

- **Administration panel** :
  Login/Password protected access. Password is encrypted in config.
  One web page to open a chatroom.
  One web page to add/remove/update moderators.
  One web page to update/close/monitor chatrooms.
  One web page to backup chatrooms.
  One web page to manage jChatBox parameters.
- **Multiple chatrooms** :
  jChatBox supports multiple chatrooms. You can open, close, updat transcript, backup each chatroom. You can also monitor users in e chatroom. Blacklist is available for each chatroom too.
- **Chatroom features** :
  Chatroom parameters are : name, subject, max. users, private me (allowed or not), display mode (frames, applet, flash), refresh moc clients are synchronized), refresh limit, history, languages (English Spanish, Danish, French, German, Dutch, Portuguese ...), filters (H Filter to prevent from cross scripting attacks and URL Filter to conv http://..... into hyperlink in the chatroom content) and moderator
- **Commands** :
  Users can run commands in chat messages (e.g. /help, /time, /kicl This feature is useful for moderators. See "User Guide" for all comr
- **Users management** :
  Moderator users can list users per chatroom. They can ban (definit kickoff (temporary) some users. Kicked users are transfered into a which can be cleared. Moderators can also track user's session in r Guest users are identified (nickname) but not authenticated.
- **System management** :
  System user can modify logs folder, backup filename, login, passw users timeout, license filename and others advanced parameters. Chatrooms backup is automatically done on servlet engine shutdov Chatrooms are automatically re-opened on servlet engine re-start.

Non-Technical : Chat Clients

- **Clients** :
  Basically, jChatBox provides 4 HTML/JavaScript skins : Mutlilangua classic (simple text skin), mIRC (mIRC look and feel), j-TV (graphi It also includes 4 Applet skins (Multilanguage, eXtremeSUN, Mangz Comics). Clients could be FLASH5&6 or Java/Swing Application too
- **jChatBox API** :
  jChatBox provides an API to let programmers/designers implement own client (HTML/Applet/Flash/Application) with their own design. Programmers can extend jChatBox Filters, Transcripts and Listener (to be notified about chatroom events).

Technical description :

Here is the technical description to let you know about what jCha need to run and how features are implemented.

Technical : Chat Server

- **Software** :
  jChatBox NEEDS **JSP1.0/Servlets2.1** engine to run.
  jChatBox need **JVM 1.2** or higher to run.
  jChatBox has been validated under **Tomcat 3.1**, **Tomcat 3.2.x**, **T 4.0.x**, **Tomcat 4.1.x**, **Tomcat 5** NEW , **Resin 1.2.3**, **Resin 2.1.x**, **3.x**, **JRun4**, **WebLogic 5.1**, **WebLogic 6.x**, **WebLogic 7.0**, **Serv 3.1**, **SunONE 7.0**, **Orion 1.5.2**, **Websphere 3.5.3**, **Websphere Websphere 5.0**. However, it should run under **any** JSP1.0/Servle higher) compliant engine. Note that most engines could be plugge web servers (Apache, IIS, NES, ...). As JAVA™ is "Write Once, Run Anywhere", jChatBox works under any OS supporting JAVA 2 such Win32, Solaris™, Linux, MACOS ....
- **Installation** :
  To install jChatBox you have to copy two jar files into your JSP/Ser engine. Then you have to select a client (HTML, Applet, Flash) and matching JSP files into the web server. Finally you have to copy configuration files. Read documentation to learn more.
- **Configuration** :
  jChatBox includes a default configuration that works without any modifications. However, you could modify it through **jchatbox.xm** conf/ folder and **web.xml**. Read documentation to learn more.
- **Backup** :
  Chatrooms backup is done manually or automatically on JSP/Servl engine shutdown (through destroy() method call). Backup file is a serialized file. Chatrooms are re-opened automatically on JSP/Serv engine re-start (through init() method call). No database needed.
- **Security :**
  First, let's talk about protocols. jChatBox only needs **HTTP**. It does any legacy protocol. jChatBox will work over Internet, on your Intr Extranet. As it needs only HTTP it should pass through **firewalls & proxies**.
  Second, let's talk about confidentiality. As guest users chat throug server, they can't know each other IP address so they can't be nuk SYSTEM user MODERATORS know about IP address. jChatBox coul over **HTTPS** (except in buffered-framed mode) so data could be er Third, Administration panel is protected by login/password. Passwc stored as plain text. jChatBox stores its **MD5** hash value. Administ URL could be modified by SYSTEM user.
  Finally, jChatBox provides an HTML filter to prevent from cross-scr attacks between guest users. Moreover, SYSTEM and MODERATOR can ban/kickoff user at any time.
- **Performances :**
  A load test NEW has been made under TOMCAT 4.1 + JVM 1.4 und 800Mhz-512MB. All results/graphs are available here. It should hel know **how much resources such as CPU, RAM, Bandwidth**, yo to handle many simultaneous users.
- **XML :**
  jChatBox is ready to XML. Any client, that could be local or remote communicates to jChatBox XMLConnector service through HTTP. C sends HTTP GET/POST requests to get chatroom information, to

login/logout and to chat. XMLConnector returns standard XML form responses that will be processed by the client.
Basically, client needs an XML parser to process responses. FLASH includes a good one. For APPLETS you can use a small one like Nan MinML...
Moreover, client can also use an XSLT processor if presentation is I XSL stylesheets.

- **Misc :**
  Business logic implemented in JavaBeans.
  XML configuration is based on **Xerces** parser.

Technical : Chat Clients

- **Clients** :
  Cookies are needed for session tracking. You may have problems if using URL Rewriting because of REFRESH Meta tag conflict.
  Clients could be **HTML/CSS/JavaScript**, **Applet**, **Flash5 or Java standalone Application** such as Swing. HTML/CSS/JavaScript, Ap skins provided in jChatBox have been validated under IE 4.x, IE5.x Netscape 4.x, Netscape 6.x, Netscape 7.x and Mozilla 1.x.
  JSP connector for Applets, Java Applications and Flash is now avail See XML Connector from jChatBox documentation included in downloadable package.
- **jChatBox API** :
  jChatBox provides an API to let programmers/designers implement own client (HTML/Applet/Flash/Application), their own filters and th design. jChatBox skins already implement both "top-bottom" and " top" display.

Download :

Download jChatBox v2.6 NEW full package (1.9 MB
(JSP + Binaries + Documentation + API)

	English	Spanish
V2.6 NEW	[zip] [tar.gz] [war|ear]	N/A
	[Mirror AT] [Mirror FI] [Mirror RU] [Mirror BR]	
V2.5	[zip] [tar.gz] [war|ear]	N/A
V2.4.1	[zip] [tar.gz] [war]	N/A
V2.3	[zip] [tar.gz] [war]	N/A
V2.2.1	[zip] [tar.gz]	N/A
V2.1.1	[zip] [tar.gz]	[zip]

jChatBox is **free for non-commercial** sites only, if you're running a cor web site then you have to purchase a license to use **here**.

---

jChatBox 2.x FREE FOR NON-COMMI

Java Desktop System

☐ Features & Benefits
» Data Sheet
» FAQ
» Detailed View

Get the Software »

Java POWERED

Table of Contents

» **Contact Me**

» Print-friendly P

Evaluate
» Product Home
» Analyst Report
» Savings Calcul
» Software from
» White Papers
» News & Event

Get
» Get the Softwa
» Support & Ser

Use
» Technical FAQ
» Documentatior
» Developers
» Communities &

END USER FEATURES

Feature	Function	Benefit
Java System Update Services	▪ Automated software updates and patching, including dependency checking/analysis to ensure systems are secure and up-to-date. ▪ Local and remote management capabilities provide update scheduling and activity log access. ▪ Interoperability with centralized software management tools.	Secure, automated and simplified software updates and patching
Tightly integrated and tuned desktop client	▪ Desktop applications are tightly integrated allowing for drag and drop and copy/paste of text, graphics and other elements between applications. ▪ Universal printer selection of available printers on network, including printers available on Windows networks. ▪ Single point of entry for directory and file management and networked computers. ▪ Direct access to content in data and files by single-click application launch for files contained in email, web pages and file manager.	Maximizes end user productivity.
Easy to Use and Learn	▪ Intuitive, easy to use, single install mechanism which ensures users can quickly get productive. ▪ Easy to learn and use, especially for	Maximizes ease of use and end user productivity.

	traditional MS Windows users. Featuring: ▪ Familiar desktop themes for file management and desktop configuration ▪ Standard keyboard short cut conventions ▪ Direct access to files and servers using the "Documents" and "Network Places" directories ▪ Familiar access to networked computers from single "Network Places" directory ▪ Full product documentation including Quick Start User Guide and other reference documentation.	
Full featured desktop client software	▪ Includes selection of great productivity tools including office suite, browser, email/calendar, instant messaging, project management, drawing and video conferencing applications. ▪ Multimedia-ready with essential video and audio players plus a sound recorder. Featuring Java Media Player with support for MP3 playback and a CD Player.	Comprehensive offering of useful desktop applications.
Interoperability	▪ Users can access data stored in office productivity files of Microsoft users and printers connected to existing Windows networks. ▪ Full interoperability with Microsoft Office files. ▪ Interoperability with POP3, LDAP, SMTP mail servers.	Reliable interoperability of files, data and printers.

Back to top

SYSTEM ADMINISTRATOR FEATURES

Feature	Function	Benefit
Policy based desktop configuration management	▪ Java Desktop System Configuration Manager allows system administrators to define groups of users and the policies for access rights and settings. ▪ Fine grained control of employee access rights and privileges.	Enables remote desktop lockdown to provide considerable savings in desktop administration and help desk calls
Centralized desktop deployment	Remote image deployment to manage a defined set of packages or system images for rapid setup and provisioning of desktops and deploy on multiple systems simultaneously.	▪ Quick desktop provisioning and deployment ▪ Automates repetitive tasks, limiting manual errors ▪ Manage, store and deploy group defined gold images
Software management tools	▪ Centralized package management enabling volume software push, patching, updating multiple desktops. ▪ Software dependency checking and analysis to identify patch and package dependencies and deploy correct set of software.	▪ Quickly deploy software, patches from multiple sources to desktops. ▪ Increase availability of desktops and keep them current and secure. ▪ Increase system administrator productivity.
Inventory and Monitoring	▪ Inventory collection of desktop details, OS, hardware profiles and more. ▪ Performance Monitoring of deployed desktop environments.	System administrator can easily see current system and health.
Grid-Enabled	Use a collection of interconnected computers as a unified computing resource.	More efficient usage of desktop resources and job scheduling.
Remote Desktop	Ability for administrator to view	More efficient help

Takeover	and interact with user's desktop display to help, guide and troubleshoot.	desk

Back to top

DEVELOPER FEATURES

Feature	Function	Benefit
Complete Developer Environment	▪ Java Studio Standard is a powerful yet intuitive Integrated Development Environment (IDE) for Java, providing a comprehensive set of features and functionality. ▪ NetBeans IDE 3.6 delivers tightly integrated deployment and debugging of web applications on Apache Tomcat 5 and the Sun Java System Application Server, Platform Edition 8. ▪ J2SE hosts the Linux GTK+ look and feel support in JFC/Swing enables Java applets and applications to be written so they look just like native Linux apps on JDS.	▪ Writing Java applets and applications for Linux ensures that those same applications will run across multiple operating systems and even multiple versions of the same O/S. Java has very strong compatibility. While the underlying Linux O/S changes regularly, Java can shield developers from those changes. ▪ Enables the development of applications ranging from desktop to standards based enterprise class applications and web services. ▪ Improves developer productivity through the optimization of the develop, execute, debug and deploy cycle. ▪ New with J2SE is a Linux GTK+ look and feel support in JFC/Swing this enables Java applets and applications to be written so they look just like native Linux apps on

JDS.

Back to top

1. This page meets my need.	disagree	○	○	○	○	○	○	agree
2. This page was easy to find.	disagree	○	○	○	○	○	○	agree


FEATURED NET TALK:
solaris